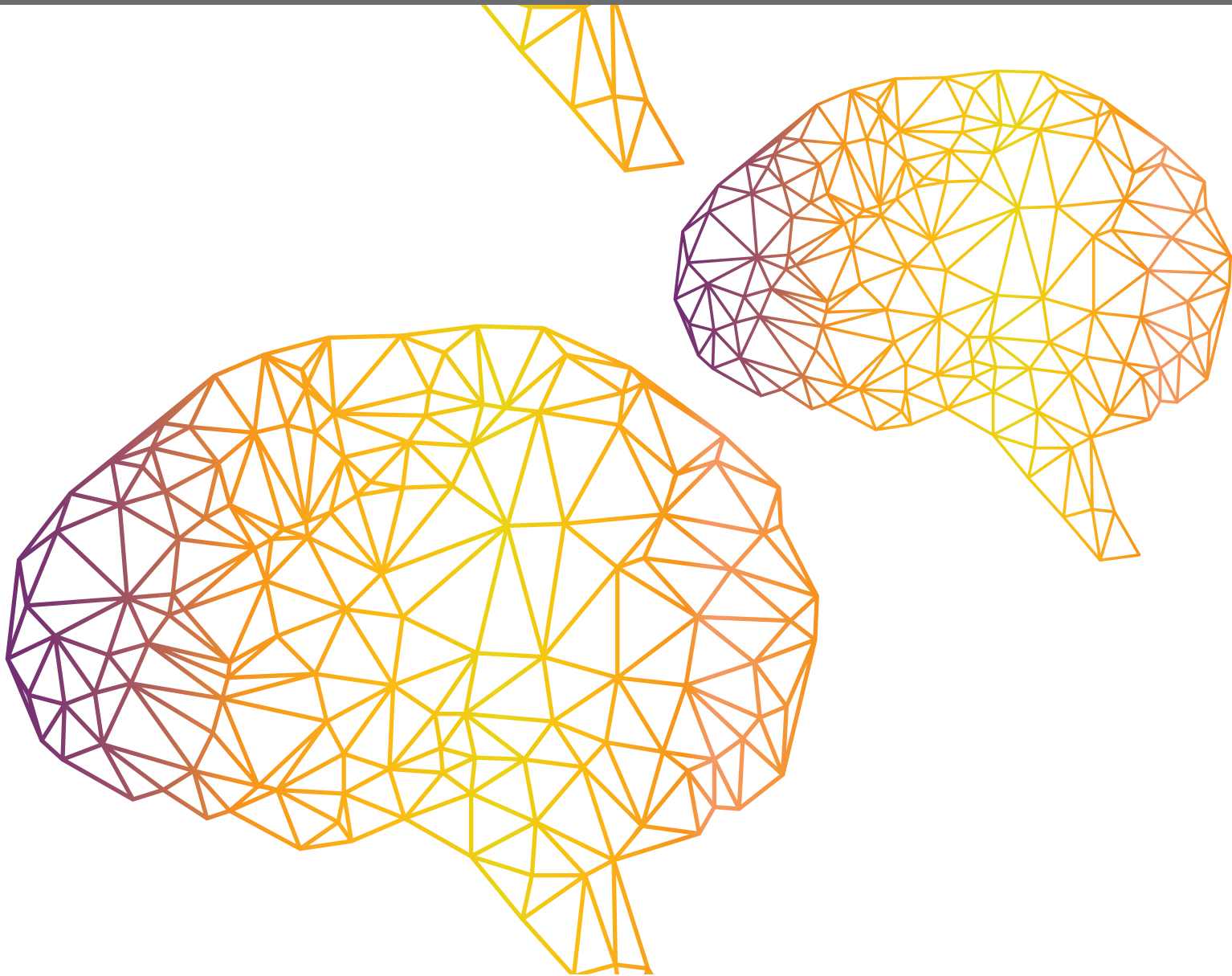# INTEGRATED MULTI-MODAL AND SENSORIMOTOR COORDINATION FOR ENHANCED HUMAN-ROBOT INTERACTION

EDITED BY: Bin Fang, Cheng Fang, Li Wen and Poramate Manoonpong

frontiers Research Topics

About Frontiers

Frontiers is more than just an open-access publisher of scholarly articles: it is a pioneering approach to the world of academia, radically improving the way scholarly research is managed. The grand vision of Frontiers is a world where all people have an equal opportunity to seek, share and generate knowledge. Frontiers provides immediate and permanent online open access to all its publications, but this alone is not enough to realize our grand goals.

Frontiers Journal Series

The Frontiers Journal Series is a multi-tier and interdisciplinary set of open-access, online journals, promising a paradigm shift from the current review, selection and dissemination processes in academic publishing. All Frontiers journals are driven by researchers for researchers; therefore, they constitute a service to the scholarly community. At the same time, the Frontiers Journal Series operates on a revolutionary invention, the tiered publishing system, initially addressing specific communities of scholars, and gradually climbing up to broader public understanding, thus serving the interests of the lay society, too.

Dedication to Quality

Each Frontiers article is a landmark of the highest quality, thanks to genuinely collaborative interactions between authors and review editors, who include some of the world's best academicians. Research must be certified by peers before entering a stream of knowledge that may eventually reach the public - and shape society; therefore, Frontiers only applies the most rigorous and unbiased reviews.
Frontiers revolutionizes research publishing by freely delivering the most outstanding research, evaluated with no bias from both the academic and social point of view. By applying the most advanced information technologies, Frontiers is catapulting scholarly publishing into a new generation.

What are Frontiers Research Topics?

Frontiers Research Topics are very popular trademarks of the Frontiers Journals Series: they are collections of at least ten articles, all centered on a particular subject. With their unique mix of varied contributions from Original Research to Review Articles, Frontiers Research Topics unify the most influential researchers, the latest key findings and historical advances in a hot research area! Find out more on how to host your own Frontiers Research Topic or contribute to one as an author by contacting the Frontiers Editorial Office: frontiersin.org/about/contact

INTEGRATED MULTI-MODAL AND SENSORIMOTOR COORDINATION FOR ENHANCED HUMAN-ROBOT INTERACTION

Topic Editors:
Bin Fang, Tsinghua University, China
Cheng Fang, University of Southern Denmark, Denmark
Li Wen, Beihang University, China
Poramate Manoonpong, University of Southern Denmark, Denmark

Table of Contents

frontiers
in Neurorobotics

Editorial: Integrated Multi-modal and Sensorimotor Coordination for Enhanced Human-Robot Interaction

Bin Fang¹*, Cheng Fang², Li Wen³ and Poramate Manoonpong⁴,⁵

¹ Department of Computer Science and Technology, Beijing National Research Center for Information Science and Technology, Tsinghua University, Beijing, China, ² University of Southern Denmark Robotics, The Maersk Mc-Kinney Moller Institute, University of Southern Denmark, Odense, Denmark, ³ School of Mechanical Engineering and Automation, Beihang University, Beijing, China, ⁴ Embodied Artificial Intelligence and Neurorobotics Lab, University of Southern Denmark Biorobotics, The Maersk Mc-Kinney Moller Institute, University of Southern Denmark, Odense, Denmark, ⁵ Bio-Inspired Robotics and Neural Engineering Lab, School of Information Science and Technology, Vidyasirimedhi Institute of Science and Technology, Rayong, Thailand

The Editorial on the Research Topic

Integrated Multi-modal and Sensorimotor Coordination for Enhanced Human-Robot Interaction

With the widespread application of intelligent robots and all kinds of sensors in our lives, human-robot interaction has become a significant research direction. Traditional human-robot interaction is often based on a single modality through a few sensors. However, with the increasing complexity of application scenarios, it is difficult to improve the interaction performance based on a single modality. Therefore, it is necessary to study human-robot interaction based on multi-modal approaches.

Integration of the sensory and motor systems allows human beings to make full use of sensory information to take meaningful motor actions. Multi-modality is often needed to gather various types of sensory information to better understand the context of complicated tasks in an ever-changing environment; for example, a cognitive or physical human-robot interaction scenario. Integrated multi-modal and sensorimotor coordination are therefore crucial for humans in tackling complex tasks requiring interactions or collaborations between humans and robots, as can be increasingly observed in many different domains, such as industry, healthcare, and rehabilitation.

The new generation of robots are meant to gradually participate in our lives and coexist in human living environments. This has encouraged investigation into technologies enabling effective interactions between humans and robots. The goal of human-robot interaction research is to make robots capable of operating in human-centered spaces to enhance work efficiency, introduce flexibility and adaptability in solutions, and improve the quality of human life.

Investigating the underlying mechanisms of multi-modal sensorimotor integration and coordination in humans will provide insight into the adaptability and compliance of human intelligence with biomechanical sensorimotor control. It will also advance the development of an equivalent robot partner through the transfer and deployment of knowledge learned from humans toward enhanced human-robot interaction. In this challenging scenario, both human and robot partners are expected to perceive their own state and that of their partner during interaction in a multi-modal manner. This involves observing the posture of the partner, monitoring the forces transmitted through a commonly operated object, communicating verbally to understand the intention of the partner, adjusting the collaboration to accomplish the task in an ergonomic and efficient way, and assigning the task roles according to the cognitive and physical strengths of the human and robot partners.

This special issue contains 16 research articles that fourteen research articles focusing on human-robot tasks, one on the object recognition and one on the systematic review.

He and Mathieu studied the extraction of different signals from a single muscle, with the potential to control the origin of multiple degrees of freedom in modern upper limb electromyography (EMG) prosthesis, demonstrating that the characteristics of biceps muscle synergy may promote the control of upper limb EMG prosthesis. In order to improve the quality of life in patients with severe dyskinesia, a brain-computer interface (BCI) for manipulator control was studied by Zhu et al., who proposed an asynchronous hybrid BCI with the ability to complete complex manipulator control tasks.

Wen et al. proposed a feature classification method based on visual sensors in dynamic environments. For detecting objects, a double projection error algorithm is proposed that combines texture and region constraints to achieve accurate feature classification in four different environments. The algorithm can classify static and dynamic feature objects and optimize the conversion relationship between frames only through visual sensors.

With the aging population and consequent increase in hemiplegic patients due to accidents, the provision of rehabilitation training has become a meaningful topic. Bong et al. developed a novel robotic system with a muscle-to-muscle interface to enhance the rehabilitation of post-stroke patients. The system can customize and adjust the rehabilitation training according to the different stages of motor recovery in stroke patients and can run in three different modes, allowing passive and active exercise for effective rehabilitation training. Zuo et al. proposed a structure of wearable parallel mechanisms with sufficient motion isotropy, high force transfer performance, and large maximum torque performance to cover all possible motion ranges in the human ankle joint complex, making it suitable for ankle joint rehabilitation. In terms of rehabilitation training equipment, the exoskeleton has been studied by many researchers. Peng et al. proposed a new data-driven optimal control strategy for adapting to the unpredictable disturbance of different hemiplegic patients. Fang et al. proposed a temporal convolutional network based gait recognition and prediction model to recognize and predict the actions of the exoskeleton wearer. Shi et al. first linearized and discretized the constraint conditions of model predictive control by using a third-order Taylor-type numerical differential formula, extending it to a lower limb rehabilitation robot to realize human-computer interaction control and intention recognition in active rehabilitation training.

Jin et al. proposed a container target recognition framework based on acoustic signals in an open environment using the kernel k-nearest neighbor algorithm. The dynamic contact method was used to collect the acoustic signal in the container to solve the problem of object recognition (Jin et al.).

Jiao et al. combined the attitude estimation system with an intelligent control structure to make the unmanned aerial vehicle (UAV) perform the detection task stably, and proposed an intuitive end-to-end interaction system, which could control the UAV according to the natural posture of the human body.

In order to make robot learning surpass human demonstration and task completion under unknown conditions, Cao et al. proposed the evolutionary strategy gradient. Through goal-oriented exploration, robot learning skills were extended to cover different parameter environments.

Li et al. proposed a multi-modal incremental learning framework based on the teleoperation strategy to reduce the error between the reconstructed and expected trajectories, enabling the robot to accurately reproduce the demonstration task.

Duque-Domingo et al. proposed a novel method for deciding who the robot should pay attention to when interacting with multiple people during the process of interaction. The method is based on the receipt of different stimuli by a competitive network (see, say, pose, hoard talk, habituation, etc.), and then competing with each other to decide who to focus on (Duque-Domingo et al.).

The article by Navarro-Alarcon et al. presents a new scheme for approximating unknown sensorimotor robot models by using feedback signals only.

Cherubini et al. systematically reviewed the existing sensor-based control methods, and then discussed the problems, potential applications, and future research directions.

Ergonomics have a significant impact on productivity as well as the chronic health risks caused by inappropriate work postures and conditions. Peternel et al. proposed a new method for estimating and transferring ergonomic working states called a Binary Work-Condition Map to provide visual feedback on the working state of different arm structures. As well as combining the advantages of both the binary map and continuous map, these researchers proposed a Hybrid Work-Condition Map for ruling out unsuitable workspaces using the binary map approach, while rendering suitable workspaces by applying the continuous map approach (Peternel et al.).

All these methods help to improve the performance of human-robot interaction. The studies demonstrate the significant potential of combining machine-learning methods and sensor technology to visualize and interpret data, ultimately enhancing the ability of human and robot cooperation to complete related tasks.

AUTHOR CONTRIBUTIONS

FUNDING

Science and Technology) - Siemens Ltd., China Joint Research Center for Industrial Intelligence and Internet of Things.

Open-Environment Robotic Acoustic Perception for Object Recognition

Shaowei Jin[1,2], Huaping Liu[3]*, Bowen Wang[1,2] and Fuchun Sun[3]

[1] *State Key Laboratory of Reliability and Intelligence of Electrical Equipment, Hebei University of Technology, Tianjin, China,* [2] *Key Laboratory of Electromagnetic Field and Electrical Apparatus Reliability of Hebei Province, Hebei University of Technology, Tianjin, China,* [3] *Department of Computer Science and Technology, Tsinghua University, Beijing, China*

Object recognition in containers is extremely difficult for robots. Dynamic audio signals are more responsive to an object's internal property. Therefore, we adopt the dynamic contact method to collect acoustic signals in the container and recognize objects in containers. Traditional machine learning is to recognize objects in a closed environment, which is not in line with practical applications. In real life, exploring objects is dynamically changing, so it is necessary to develop methods that can recognize all classes of objects in an open environment. A framework for recognizing objects in containers using acoustic signals in an open environment is proposed, and then the kernel k nearest neighbor algorithm in an open environment (OSKKNN) is set. An acoustic dataset is collected, and the feasibility of the method is verified on the dataset, which greatly promotes the recognition of objects in an open environment. And it also proves that the use of acoustic to recognize objects in containers has good value.

Keywords: open environment, interactive perception, objects in containers, acoustic features, object recognition, kernel k nearest neighbor

1. INTRODUCTION

With the development of intelligent robots and artificial intelligence, the demand for intelligent service robots in society is increasing. For example, intelligent service robots can take care of the elderly. But intelligent service robots live with older people and must have the same perceptual abilities as people, such as being able to see, touch, or hear what is happening in the world around them. These perceptual abilities will enable robots to perform various tasks, among which object recognition is one of the most common and important tasks. To accomplish this task, intelligent service robots can be equipped with multiple types of sensors, each of which can reflect the properties of an object from different aspects. Currently, the most widely used sensor is the camera, because a large amount of information about an object can be obtained from a single image. Therefore, the study of vision in object recognition has attracted great attention. For example, color (Forero et al., 2018), texture (Kaljahi et al., 2019), and appearance (Liu et al., 2010) can be classified by visual images. However, the vision is sometimes affected by factors such as illumination, object color, occlusion, and posture of objects. It is difficult to find some intrinsic properties of objects, such as softness, stiffness, and material properties.

In addition, the force sensor responds to some object properties according to the contact force when contacting the object, for example, by directly contacting the object for shape recognition (Luo et al., 2016), category recognition (Gandarias et al., 2018), material retrieval (Strese et al., 2017), and surface roughness recognition (Yi et al., 2017). By simply touching an object that only recognizes the object being touched, it is impossible to perceive objects in the container, such

as food in a kitchen container or medication in a bottle. Interactive perception is a common human exploration strategy. If humans cannot recognize objects by vision and touch, they will take different interactions to obtain information about other sensory channels. For example, shaking a hollow object produces auditory information that can determine whether the object is empty or full, what material is contained inside, and how much material is inside.

Sound-based object recognition is less studied than object recognition based on visual and tactile information. Despite this, hearing is as important as touch and vision, especially in the dark or dangerous environments. Hearing has a unique advantage. Sound and structural vibration signals provide a rich source of information for manipulating objects. Humans use this feedback to detect mechanical events and estimate the state of the manipulated object. Hearing allows us to infer events in the world that often go beyond the scope of other sensory modes. Studies have shown that humans are able to extract the physical properties of objects and distinguish between different types of events from the sound produced (Grassi, 2005; Beran, 2012). There has been some research that uses sound for positioning (Brichetto et al., 2018), cup material recognition (Griffith et al., 2012), pouring height (Liang et al., 2019), family event recognition (Chang et al., 2016; Do et al., 2016), object material recognition (Neumann et al., 2018), contact position recognition and qualitative size of contact force (Zöller et al., 2018).

The ability to sense and process vibrations during interactive contact with an object will allow the robot to detect anomalies in the interaction process and perform object recognition based on the vibration signals. The interaction between objects produces a vibration signal that propagates through the air and can be perceived by the acoustic sensor. The cost of collecting and processing vibration feedback is relatively low relative to other sensory modes (such as vision). Object recognition is a fundamental skill that occurs during the early development of human beings. When interacting with objects, it will try a variety of interactive ways to complete the task of recognizing objects. At the same time, when contacting an object, it is necessary to detect whether the object in contact has been encountered before, that is, the object being explored is an object of unknown classes. Making a robot separate objects of unknown classes from objects of known classes like a human, then relearn the information of the unknown classes. Therefore, it is very important for robots to develop a system that can recognize objects of unknown classes and reach the recognition of all objects by continuously learning the properties of unknown classes. The main contributions of this paper are as follows:

1. A framework for recognizing objects in containers using acoustic signals in an open environment is proposed.
2. The kernel k nearest neighbor algorithm (OSKKNN) in an open environment is proposed to solve the problem of recognizing all class objects in an open environment.
3. The acoustic dataset was collected using the UR5 arm with the microphone and verified the effectiveness of our method.

The remainder of the paper is organized as follows. In section 2, the related work of collecting sound to recognize objects is reviewed. In section 3, a framework for recognizing objects using sound is introduced, and an OSKKNN algorithm is proposed. Acoustic dataset collection and data analysis and processing are in section 4. Section 5 conducts experiments and experimental analysis. Finally, the paper is concluded in the last section.

2. RELATED WORK

It is very challenging to recognize objects contained in containers and objects of different weight in containers. These studies are few. When the visual and tactile constraints are limited, the perceptual information generated by the simple static contact is also difficult to recognize objects in containers. We naturally use dynamic contact methods to obtain information about objects. Berthouze et al. (2007) and Takamuku et al. (2008) pointed out that dynamic contact (shaking) is more conducive to recognizing objects than static contact (grasping), which is not easily affected by the shape, size and color of objects. When shaking the object, it will produce the sound signal of vibration, which can be collected by a microphone. There is related research on the use of shaking to collect sound signals (Nakamura et al., 2009, 2013; Araki et al., 2011; Taniguchi et al., 2018).

Interactive contacts with objects in different ways and acquisition of sound information to recognize objects are studied as follows: Clarke et al. (2018) used the actions of shaking and pouring to obtain the sound signal of the granular object and combined this with deep learning to recognize five different types of granular objects. Luo et al. (2017) used a pen to hit objects to collect sound information, and used the Mel-Frequency Cepstral Coefficients (MFCCs) and its first and second differential as features; stacked denoising autoencoders are applied to train a deep learning model for object recognition. Sinapov et al. (2009) and Sinapov and Stoytchev (2009) used humanoid robots to perform five different interactive behaviors (grasp, shake, put, push, knock) on 36 common household objects (such as cups, balls, boxes, cans, etc.) and used the k nearest neighbor algorithm (KNN), support vector machine algorithm (SVM) and unsupervised hierarchical clustering to recognize objects. Sinapov et al. (2011) collected the joint torque of the robot and sound signals, and combined with the k nearest neighbor algorithm (KNN) to recognize 50 common household objects.

Sinapov et al. (2014) and Schenck et al. (2014) used ten kinds of interactions (such as grasp, shake, push, lift, etc.) to detect four classes of large-particle objects of three colors and three weights. They not only learn categories describing individual objects, but also learn categories describing pairs and groups of objects, and the C4.5 decision tree algorithm is used to classify and the robot learns new classes based on the similarity measurement method. Chen et al. (2016) tested four kinds of containers (glass, plastic, cardboard and soft paper) with 12 kinds of objects and collected sound signals through shaking using Gaussian naive Bayes algorithm (GNB), support vector machine algorithm (SVM) and K-means clustering algorithm (K-Means) to classify and recognize objects. And it is proved that

FIGURE 1 | Acoustic recognition framework in open environment.
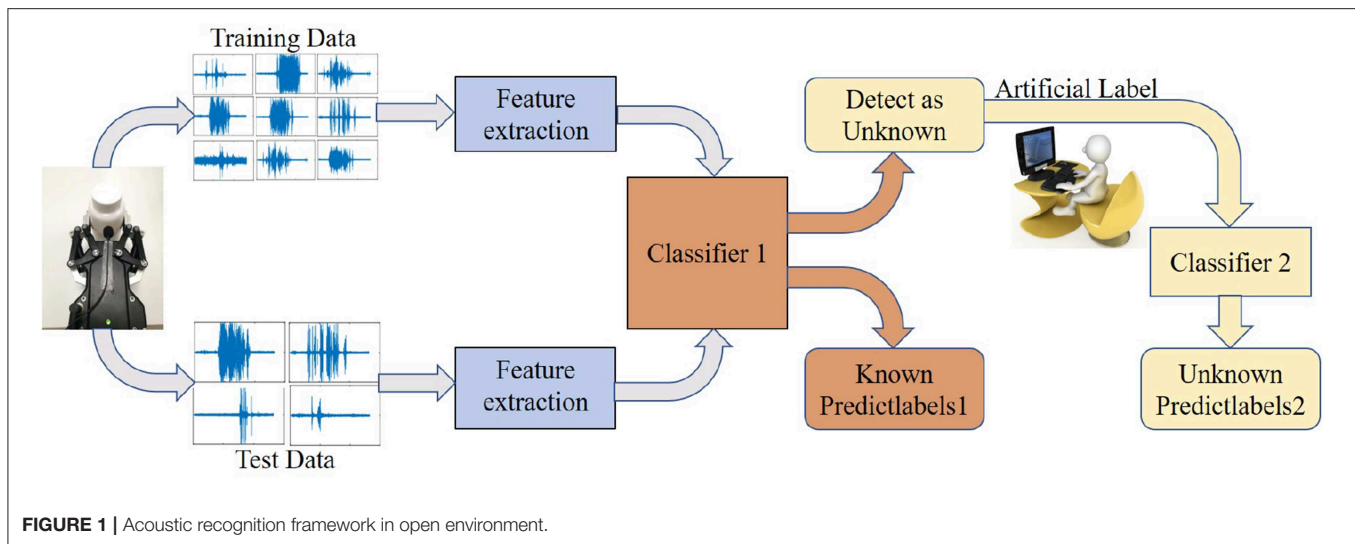
the sound of shaking can be used for object recognition in many places such as shopping malls, workshops and home. Eppe et al. (2018) used a humanoid robot to perform auditory exploration of a group of visually indistinguishable plastic containers filled with different amounts of different materials, proving that deep recursive neural structures can learn to distinguish individual materials and estimate their weight.

The above research focuses on object recognition in multiple occasions and closed environments and does not pay attention to recognizing objects in specific applications and open environments. There are some studies on different recognition in the open environment, such as the following literature (Bendale and Boult, 2016; Bapst et al., 2017; Gunther et al., 2017; Moeini et al., 2017; Bao et al., 2018), these studies recognize known classes and detect unknown classes in an open environment, but do not recognize all classes. In the real world, objects touched by robots are constantly changing. How can the robot system be made to be like human beings? When encountering unknown objects, it can be well separated from known objects and relearn relevant knowledge of unknown objects. Therefore, it is very important for robots to develop a systematic framework that can detect objects of unknown classes and recognize all objects through continuous learning of the properties of unknown classes. This paper mainly studies the use of sound to recognize household food objects in containers and focuses on the recognition of all class objects in containers using sound in an open environment.

3. KERNEL K NEAREST NEIGHBOR METHOD FOR ACOUSTIC RECOGNITION IN OPEN ENVIRONMENT

The open environment no longer assumes that the test set classes and the training set classes are identical, and the open environment is more in line with the actual process of exploring objects. The open environment points out that the classes of

the test set will present classes never seen on the training set. Traditional machine learning is carried out in a closed environment and cannot solve the object recognition problem in the actual open environment. If there are unknown classes in the test set, the objects of these unknown classes will be marked as some classes in the training set by using traditional machine learning, which is not in line with the actual classification recognition process and human learning process. This paper develops a systematic framework for recognizing objects using audio in an open environment.

3.1. Open Environment Acoustic Recognition Framework

In the actual learning process, people can recognize the object well when they encounter an object that needs to be recognized. On the other hand, even if the people do not know it they will say that this is an unknown class that they had not seen. If the people want to recognize what this object is, they must recognize it by looking up a book or asking someone who recognizes it to relearn. The system framework is proposed in this paper, which uses audio to recognize objects in an open environment. It is similar to the human learning process. The system framework is shown in **Figure 1**.

The acoustic recognition process in an open environment is shown in **Figure 1**. Under this framework, the acoustic data are collected by the robot platform, and the data set is divided into a test set and a training set, but the test set contains classes not seen in the training set. First, acoustic feature extraction is performed on the data, and then the classifier 1 recognizes known object classes of a test set through training a training set, and the classifier 1 detects unknown object classes of a test set. These data of unknown classes are collected and manually labeled. Finally, these data of unknown classes are relearned and trained by the classifier 2. The key part in the acoustic recognition framework is the role of classifier 1. It not only needs to recognize accurately objects of known classes but also needs to detect

objects of unknown classes. Only objects with unknown classes can be detected for better relearning and recognizing.

3.2. Kernel k Nearest Neighbor in an Open Environment

The k nearest neighbor algorithm is the most mature and simplest classification algorithm, with low training time complexity. It is widely used in various fields, such as text classification (Yong et al., 2009), face recognition (Weinberger et al., 2006), image classification (Zhang et al., 2006; Boiman et al., 2008; Guru et al., 2010), object recognition (Sinapov et al., 2009, 2011), etc. In order to solve different problems, relevant research has been done to improve and optimize the k nearest neighbor algorithm, as in the literature (Weinberger et al., 2006; Zhang et al., 2006; Boiman et al., 2008; Yong et al., 2009; Guru et al., 2010; Kibanov et al., 2018; Liao and Kuo, 2018). This paper uses acoustic properties to recognize objects. The properties of these objects are very close, such as particle size, density, so collected acoustic data are linearly inseparable. If the training test data set has the problem of linear inseparability, then the k nearest neighbor algorithm's similarity measurement effect dependent on distance will become worse and the recognition accuracy will be reduced. To solve this problem, kernel function can be introduced into the k nearest neighbor algorithm to improve the recognition effect of the nearest neighbor algorithm (Yu et al., 2002).

Traditional machine learning is classified and recognized in a closed environment. The open environment recognition problem is a class that does not exist in the training sample in the test sample. It is very difficult to use the kernel k nearest neighbor algorithm in an open environment for this problem. Therefore, based on the kernel k nearest neighbor algorithm, this paper proposes OSKKNN algorithm to recognize objects of all classes in an open environment.

Kernel function includes linear kernel function, polynomial kernel function and radial basis kernel function. Among them, the radial basis kernel function is the most commonly used kernel function, which can map data to infinite dimensions and is a scalar function with radial symmetry. In this paper, the radial basis kernel function is introduced into the k nearest neighbor algorithm.

Definition of radial basis function: X_1 and X_2 represent the eigenvectors of input space, the radial basis kernel function is as follows:

$$K(X_1, X_2) = \exp\left(-\frac{(\|X_1 - X_2\|)^2}{2\sigma^2}\right) = \exp(-\gamma(\|X_1 - X_2\|)^2) \quad (1)$$

Where σ is the hyperparameter of RBF kernel, and the characteristic length-scale of learning samples' similarity is defined, that is, the proportion of the distance between samples before and after feature space mapping in the perspective of weight space (Chang and Lin, 2011; Liao and Kuo, 2018), which can be simplified into a more general form when $\gamma = \frac{1}{2\sigma^2}$.

The OSKKNN algorithm is described as follows:

(1) Convert training samples and test samples into kernel matrix representations by kernel function;

(2) Calculate the distance from the test set sample to each training set sample by the kernel matrix representation;

(3) Sort by distance from near to far;

(4) Selecting a training set sample of k closest to the current test set sample as a neighbor of the test sample;

(5) Count the class frequencies of the k neighbors;

(6) Calculate the average value of the nearest k neighbor distances, compare the size of the average value and the threshold value T;

(7). If it is less than this threshold value T, the class with the highest frequency among the k neighbors is the class of the test sample;

(8) If it is greater than this threshold value T, the test sample is of unknown classes;

(9) Collect these unknown class samples and divide them into training set and test set;

(10) Use the kernel k nearest neighbor algorithm to train and recognize these unknown objects.

4. ACOUSTIC DATASET COLLECTION

Our dataset is obtained through the robot experiment platform shown in **Figure 2**. The robot experiment platform is mainly composed of five parts: Fixed platform, UR5 robot arm (Universal Robots), AG-95 manipulator (DH Robotics), microphone (acquisition frequency 44. 1 kHz, the sound is collected through a standard 35 mm plug into the computer interface, and the sound data are read and saved using the Matlab program) and object placement table. The AG-95 manipulator and UR5 robot arm are connected structurally through a flange and communicate with UR5 through a network wire. Moreover, the grabbing experiment of objects can be easily realized through programming tools equipped with UR5. In order to reduce the impact of environmental noise, we fixed the microphone to the palm of the AG-95 manipulator. During the experiment, the UR5 robot arm drives the AG-95 manipulator to grab the container with different objects on the object placement table according to the planned path, completes the specified shaking action in the air, collects the acoustic signal during the shaking process, and then puts the container back to the original position and returns the UR5 robot arm and AG-95 manipulator to the original position.

4.1. Interaction Actions

When the robot interacts with the object, the container is shaken and the object in the container collides with the wall of the container to generate a sound waveform. The time for collecting the contact object is 6 s, and the sampling frequency is 44.1 kHz. In order to test which acquisition method is more suitable for robots and recognition, we use the following four actions, which can be implemented by the programming tool of UR5 robot arm configuration.

(1) Rotate 90°: As shown in **Figure 3A**. Turn clockwise 90° from the direction of the AG-95 manipulator's finger. At the beginning, the container mouth is up before rotating AG-95 manipulator, as shown in the position on the left
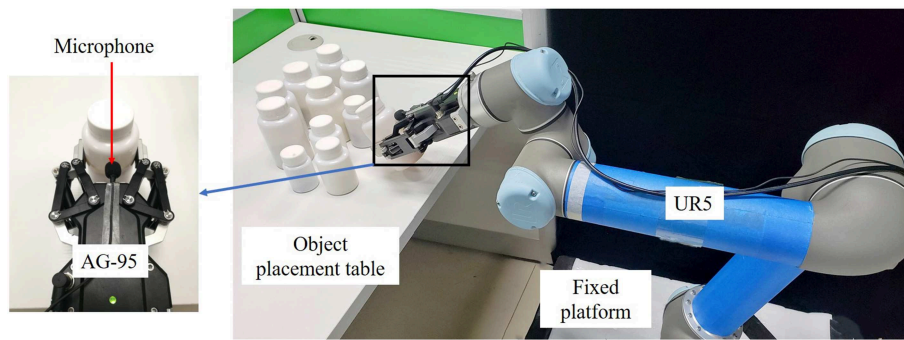
FIGURE 2 | Robot experiment platform.

A Rotate 90 degrees

B Rotate 180 degrees

C Rotate 180 degrees horizontally

D Shift from the bottom up
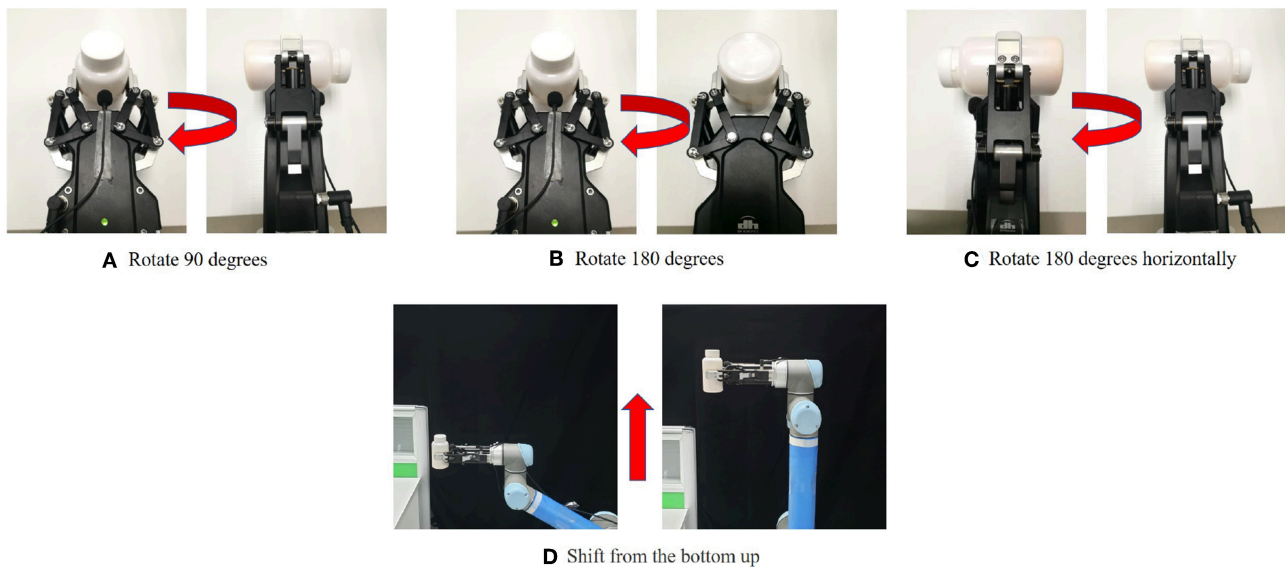
FIGURE 3 | Interaction actions.

side of **Figure 3A**. The container mouth is to the right after rotating AG-95 manipulator, as shown in the position on the right side of **Figure 3A**. Turn on the sound collection device before rotating, turn off the sound collection device after rotating, and then save the data.

(2) Rotate 180°: Same as rotate 90°, except that the angle of rotation is different. Rotate clockwise 180° from the direction of the AG-95 manipulator's finger, as shown in **Figure 3B**.

(3) Rotate 180° horizontally: As shown in **Figure 3C**. Turn 180° clockwise from the direction of the finger of the AG-95 manipulator. At the beginning, the mouth of the container is on the left before rotating the AG-95 manipulator, as shown in the position on the left side of **Figure 3C**. The mouth of the container is to the right after rotating the AG-95 manipulator, as shown in the position on the right side of **Figure 3C**. Turn on the sound collection device before rotating, turn off the sound collection device after rotating, and then save the data.

(4) Shift from the bottom up: As shown in **Figure 3D**. Move parallel from bottom to top in the air. Before moving, the AG-95 manipulator is in the lower position on the left side of **Figure 3D**, wait to open the sound collection device. After the end of the movement, the position on the right side of **Figure 3D** is at a higher position. Then turn off the sound collection device and save the data.

4.2. Objects Selections

In order to be applied to household food recognition, 20 kinds of food materials and medicines are selected as shown in **Figure 4**, and the selected objects are difficult to distinguish. Data acquisition is performed using the interactive methods of section 4.1, and each object is subjected to sound collection 30 times. The sound data are collected on a 16-bit mono at a sampling frequency of 44.1 kHz and saved as a waveform file. The sample data collected by each method are $20 \times 30 = 600$, and the sample data are collected by the four interaction methods. So, an acoustic dataset of 2,400 samples was established.
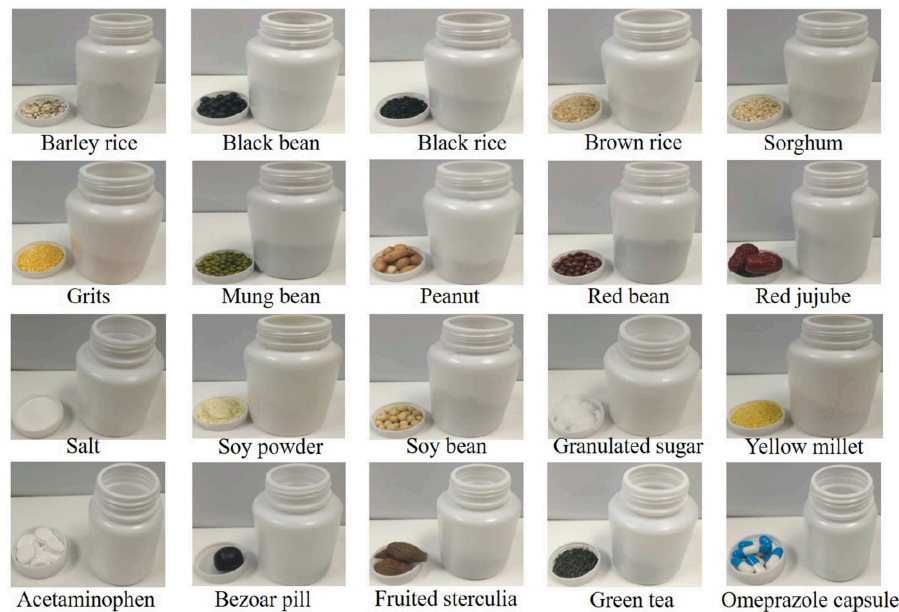
FIGURE 4 | Objects and bottles used.

Figure 4 shows different objects and corresponding plastic bottles. The plastic bottles used are made of polyethylene. The volume of the plastic bottle is as follows: 200 ml (bottom diameter: 6.1 cm, bottle height: 8.9 cm), 100 ml (bottom diameter: 5.1 cm, bottle height: 5.6 cm), 50ml (bottom diameter: 4.1 cm, bottle height: 4.4 cm). Plastic bottles of different volumes are chosen because the volume of containers required to hold different objects in real life is different.

Figure 4 shows order of the items (weight, volume of the container): barley rice (50 g, 200 ml), black bean (50 g, 200 ml), black rice (50 g, 200 ml), brown rice (50 g, 200 ml), sorghum (50 g, 200 ml), grits (50 g, 200 ml), mung bean (50 g, 200 ml), peanut (50 g, 200 ml), red bean (50 g, 200 ml), red jujube (50 g, 200 ml), salt (50 g, 100 ml), soy powder (50 g, 200 ml), soybean (50 g, 200 ml), white granulated sugar (50 g, 100 ml), yellow millet (50 g, 200 ml), acetaminophen tablet (10 g, 50 ml), bazaar supernatant pill (18 g, 100 ml), fruited sterculia (20 g,100 ml), green tea (12 g, 100 ml) and omeprazole capsule (8 g, 50 ml).

4.3. Data Analysis and Processing

The sound signals are collected by objects in the container in different interactive methods. **Figure 5** shows four representative objects and collected acoustic signals. By longitudinal comparison, it can be concluded that a sound signal can be used for recognition research. By horizontal comparison, it can be concluded that the sound signals collected by different interactive methods are different, and the recognition effects will be different. In the process of sound collection, a wiener filter needs to be firstly adopted to reduce noise due to the large noise of the environment and UR5 robot arm.

Wiener filtering is a wiener filtering algorithm based on a priori SNR proposed by Scalart and Filho (1996). It is an optimal estimator for stationary processes based on the minimum mean

square error criterion. The mean square error between the output of this filter and the desired output is minimal, so it is an optimal filtering system. It can be used to extract signals that are contaminated by stationary noise (Le Roux and Vincent, 2012).

In order to reflect better the properties of the object, it is necessary to propose better features from the noise signal after noise reduction. There are other methods of dimensionality reduction (Li et al., 2017), but the Mel-Frequency Cepstral Coefficients(MFCCs) can not only reduce the data dimension but also the dynamic properties of the sound. The MFCCs are one of the most commonly used features in speech processing. The feature extraction method can also effectively reduce the dimension, thus reducing the computational cost. Related studies have successfully applied the Mel-Frequency Cepstral Coefficients (MFCCs) to speech feature extraction and object recognition, as in the literature (Nakamura et al., 2013; Luo et al., 2017; Eppe et al., 2018). The standard MFCC feature can only propose the static characteristics of the sound (Cao et al., 2017). In order to better reflect the dynamic characteristics of the sound, this paper uses the first-order and second-order different features of the static 12-order MFCCs to obtain the dynamic features of 36-dimensional MFCCs.

5. EXPERIMENT

5.1. Object Recognition in Closed Environment

5.1.1. Comparison of Learning Algorithms and Comparison of Interaction Methods

The classification problem in a closed environment assumes that the training set and the test set have the same classes of objects. In this section, we selected four supervised learning algorithms
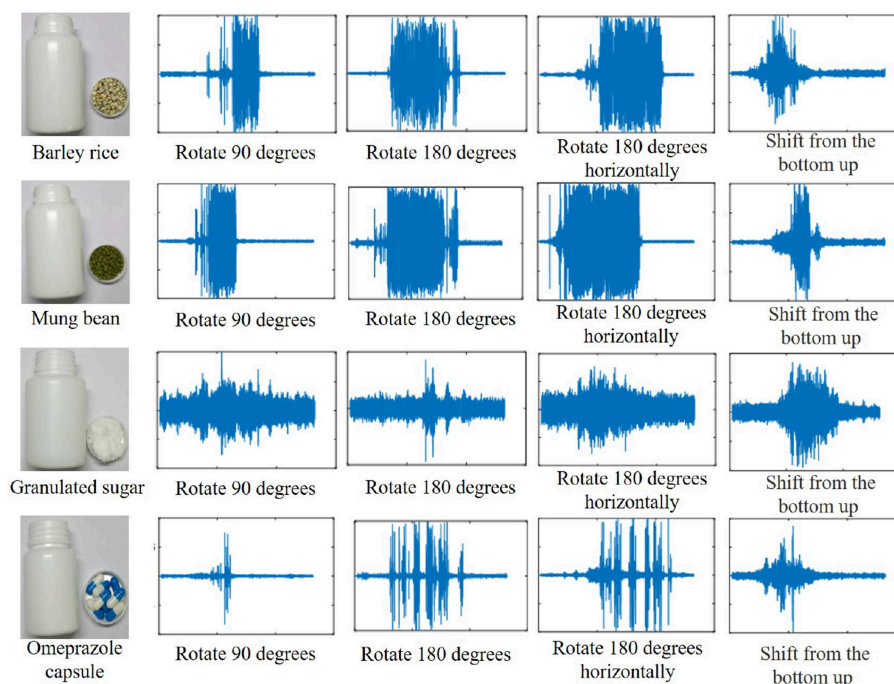
FIGURE 5 | Acoustic waveforms of four kinds of objects (Barley rice, Mung bean, Granulated sugar, Omeprazole capsules) were collected by interactive methods.

for comparison, namely the traditional k nearest neighbor algorithm (KNN), the support vector machine algorithm (SVM) of radial basis kernel function (Chang and Lin, 2011), the sparse representation classification algorithm (SRC) (Patel et al., 2011; Pillai et al., 2011), and the kernel k nearest neighbor algorithm (Kernel-KNN) (Yu et al., 2002). We first determined the K-values of KNN and Kernel-KNN through experiments. The experimental results were better in the K range of 8–16. We took $K = 14$ in OSKKNN. We used four different interactive methods when collecting data, so we conducted recognition experiments on four data sets. The proportion of the data in the training set and test set is 2:1, and the experimental results are shown in **Table 1**.

Firstly, the influence of the data interaction method on object recognition accuracy is compared with the experimental results in **Table 1**. The experimental results in **Table 1** show that the interactive method of rotating 180° horizontally has a good recognition effect on objects, with the recognition accuracy reaching 85.5%. The recognition effect of rotating 180° is close to that of rotating 180° horizontally, with a difference of 0–5% in recognition accuracy. Therefore, it is appropriate to use the wrist rotation of 180° when the robot collects sound data. By comparing rotate 90° and rotate 180°, it can be inferred that the larger the rotation angle is in the same direction, the more information about object properties is reflected in the collected data, and the higher the recognition accuracy is. Rotate 180° and rotate 180° horizontally, and rotate 90° are rotated around the wrist of the UR5 robot arm. The noise of the UR5 robot arm itself is small, so the data collection can better reflect the attributes of the object. The single joint rotation method can be used to

TABLE 1 | The recognition accuracy of different interactions and supervised learning methods.

Acquisition methods	KNN%	SVM%	SRC%	Kernel-KNN%
Rotate 90°	56	62.5	48.5	65.5
Rotate 180°	75	78.5	60.5	82.5
Rotate 180° horizontally	74.5	79.5	75	85.5
Shift from the bottom up	33	34.5	32.5	33.5

explore the attributes of the object interactively and achieve better recognition effect. The way of shifting from the bottom up, the multi-joint movement of the UR5 robot arm and the loud noise of the UR5 robot arm itself completely cover the sound of the interaction between the AG-95 manipulator and object, and the sound reflecting the object properties cannot be collected. The data set is full of noise, so the object recognition accuracy is very low.

By comparing the four supervised learning algorithms, the experimental results in **Table 1** are as follows: the kernel k nearest neighbor algorithm has the best recognition effect and is more suitable for multi-classing. Therefore, the recognition effect after combining the kernel function exceeds the support vector machine algorithm. The k nearest neighbor algorithm combined with the kernel function improves the recognition performance compared with the traditional k nearest neighbor algorithm, and better solves the problem of linear indivisible object recognition. Sparse representation classification was first proposed for image recognition, and the recognition of sound information and small

data sets is not good. Therefore, this paper extends the Kernel-KNN algorithm to solve the problem of recognizing unknown class objects in an open environment.

We consider the influence of the kernel function on the KNN algorithm. We compare the linear kernel, the polynomial kernel, the Sigmoid kernel, the rational quadratic kernel, and the radial basis kernel on a horizontally rotated 180° data set. The experimental results are shown in **Table 2**. Experiments show that the KNN algorithm with radial basis kernel is better.

5.1.2. Comparison of Weight
When the object is in constant use, the weight is decreasing. For the same object, regardless of the weight, it must recognize the same object. Therefore, we conducted the following experiments, the first interactive method (rotate 180°) is used to collect sound

TABLE 2 | Recognition accuracy of different kernel functions.

Dataset	Linear%	Polynomial%	Sigmoid%	Rational quadratic%	Radial basis%
Rotate 180° horizontally	72.5	75	71	85	85.5

TABLE 3 | The recognition accuracy of different weights and different supervised learning methods.

Weight (g)	KNN%	SVM%	SRC%	Kernel-KNN%
50	70	72	68	78
100	66	73	50	74
Mixture of 50 and 100	64.5	68	50	73

data of 10 objects for verification. The 10 kinds of objects are barley rice, black bean, black rice, brown rice, sorghum, grits, mung bean, red bean, soybean, and yellow millet; the plastic bottles used in these 10 objects are the same plastic bottles as the corresponding ones in section 4.2, but weight 100 g.

The experimental results are shown in **Table 3**. It can be concluded from **Table 3** that the Kernel-KNN algorithm has a better recognition effect when the object in the plastic bottle is 50 g. This is because the generated sound is larger when the weight of the object in the bottle is 50 g, and it can be inferred that the recognition accuracy of the internal object recognition rate decreases with the increase of weight when the bottle volume is the same. For mixed recognition of 50 and 100 g, the confusion matrix of recognition is shown in **Figure 6**. Although the recognition effect decreases by 0.5 to 5%, the object in a container can still be well recognized. It proves that no matter what the weight of the object in a container is, the object in the container can also be well recognized by using sound. At the same time, it shows that the sound has a valuable application value for a home intelligent service robot to recognize objects in containers.

5.2. Object Recognition in Open Environment
Object recognition in an open environment is that the classes of test set and training set are not the same, and the number of classes of the test set is greater than that of the training set. Object recognition in an open environment is more suitable for practical application and the human learning process. Object recognition in an open environment requires two steps. Firstly, samples of known classes are recognized in the open environment, samples of unknown classes are detected and collected in an open environment. Then, these unknown classes are re-learned, which is in line with the continuous learning process of human beings.
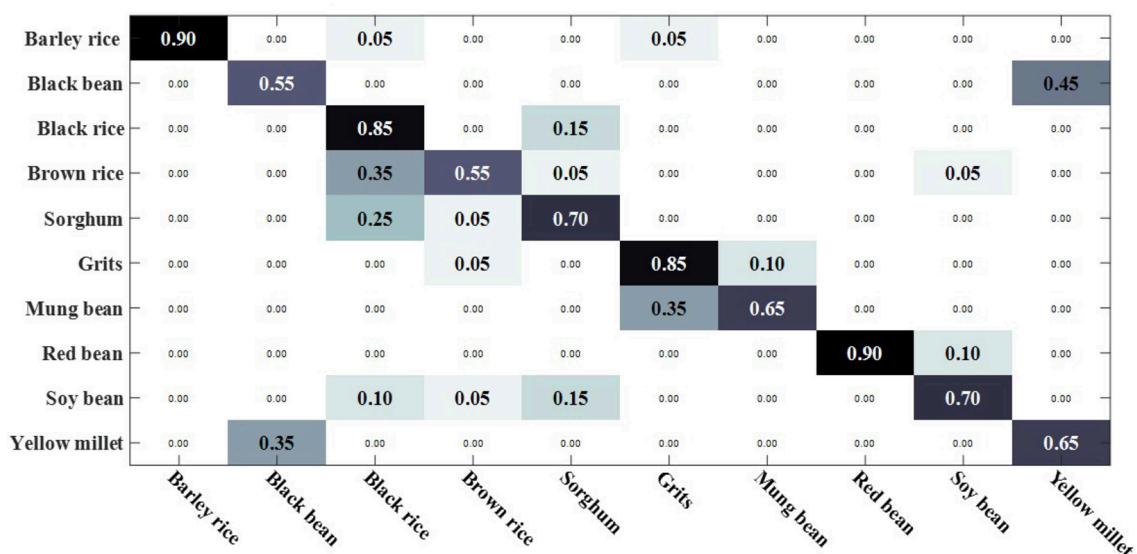
FIGURE 6 | Kernel-KNN recognition confusion matrix of mixed 50 and 100g.
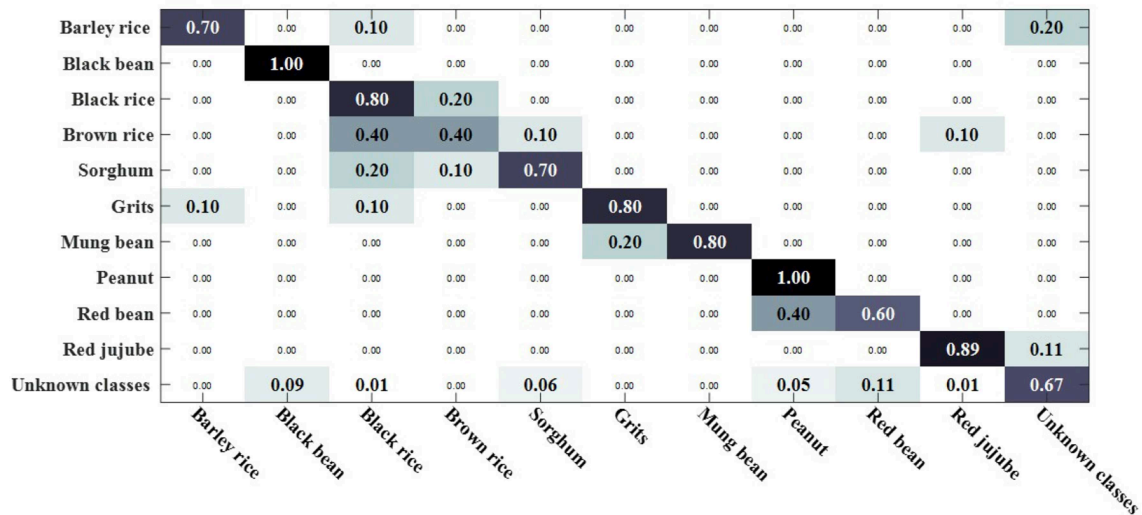
FIGURE 7 | OSKKNN recognition confusion matrix on the data of rotate 180°.
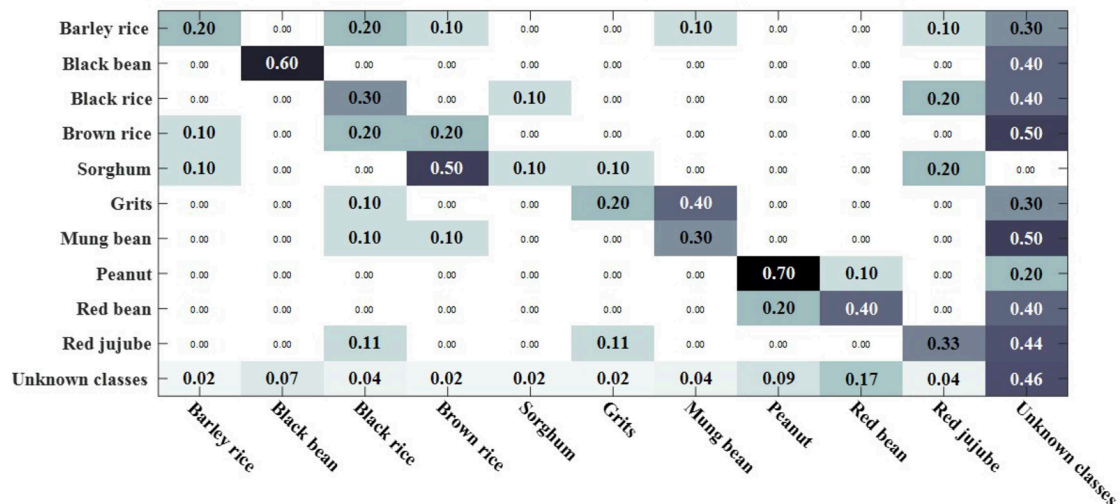
FIGURE 8 | OSSRC recognition confusion matrix on the data of rotate 180°.

5.2.1. Detection of Unknown Class Objects in Open Environment

In the open environment, the known classes are recognized and the unknown classes are detected, and then the objects of the unknown classes are collected. In other words, no matter how many classes the unknown classes are, it is not recognized as long as the unknown classes are detected.

We compared the open set sparse representation classification method for recognizing faces in an open environment (Moeini et al., 2017). This method only recognizes known classes in an open environment, detects unknown classes, and does not recognize unknown classes, so the OSKKNN algorithm is set up in the same way. The experiment was compared on a data set of rotating 180° and rotating 180° horizontally. We set up a training set of 10 classes and test sets of 20 classes (including 10 of the training set).

Figures 7, 8 are, respectively, the confusion matrix of OSKKNN and OSSRC on the dataset of rotating 180°. Figures 9, 10 are, respectively, the confusion matrix of OSKKNN and OSSRC on the dataset of rotating 180° horizontally. By comparing Figures 7–10, it can be concluded that OSKKNN has a better effect than OSSRC in separating known classes and unknown classes, and has a better effect in recognizing known classes. Only by separating sample data from unknown classes can you better collect this data and prepare to continue learning these unknown classes. As far as we know, there is
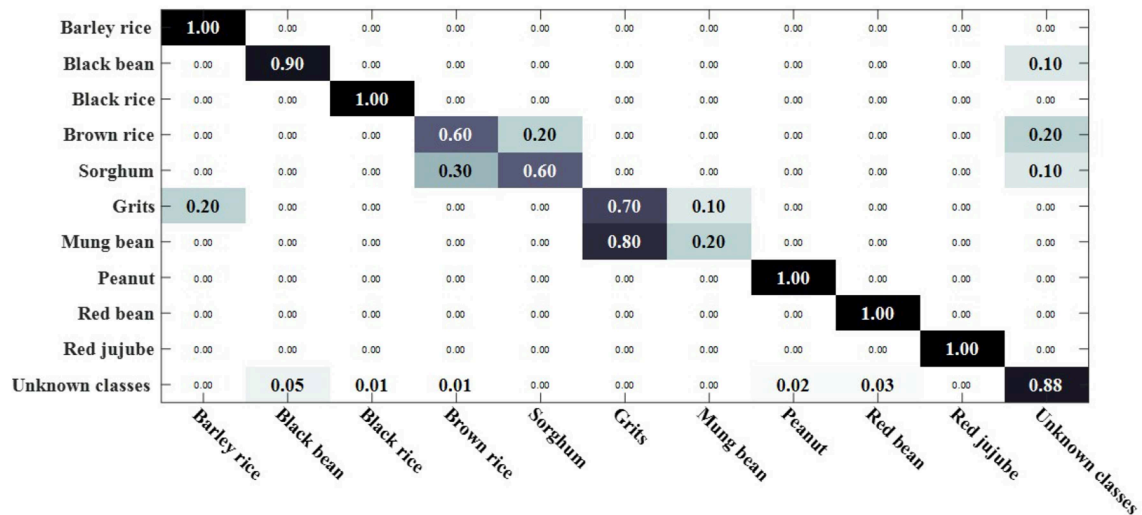
FIGURE 9 | OSKKNN recognition confusion matrix on the data of rotate 180° horizontally.
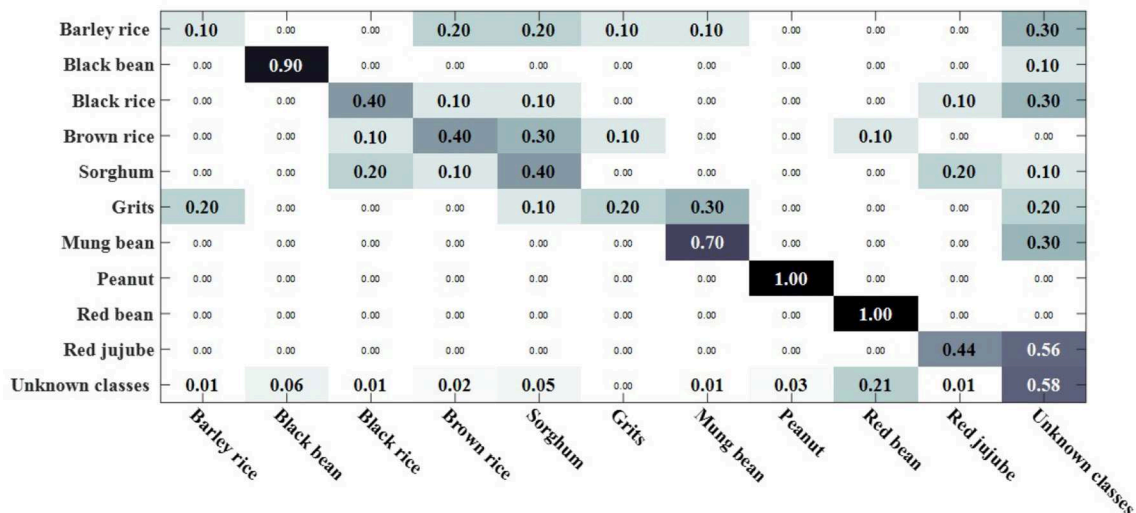
FIGURE 10 | OSSRC recognition confusion matrix on the data of rotate 180° horizontally.

no research on the object recognition of unknown classes in an open environment at present. There are only similar studies (Moeini et al., 2017) that recognize objects of known classes in the open environment and detect unknown classes without recognizing objects of unknown classes. OSKKNN can recognize objects of unknown classes detected, and finally recognize objects of all classes.

5.2.2. Recognition of All Class Objects in Open Environment

The threshold value T is an important factor affecting the recognition of all class objects in an open environment. A reasonable threshold value T setting can successfully separate objects of known classes from objects of unknown classes.

The influence of unknown classes on the recognition accuracy of known classes can be reduced and the overall recognition accuracy can be increased only by separating known classes from unknown classes in the test set.

The threshold value T selection is different for different dataset thresholds and needs to be determined experimentally. We experimented on rotate 180° data and rotate 180° horizontally data. The number of classes in the experimental training set was randomly selected from 10 classes. The test classes included all classes.

The experimental results are shown in **Figures 11, 12**. It can be seen from **Figures 11, 12** that the recognition accuracy of the known classes is decreasing as the threshold is increased. This is because the larger the threshold value T is, the more samples
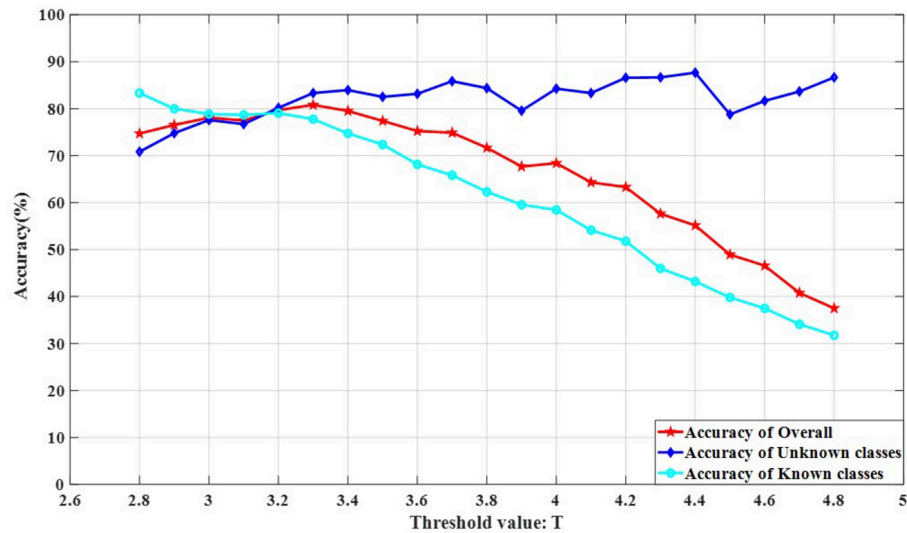
FIGURE 11 | The influence of OSKKNN threshold value *T* on the recognition accuracy of rotate 180° data.
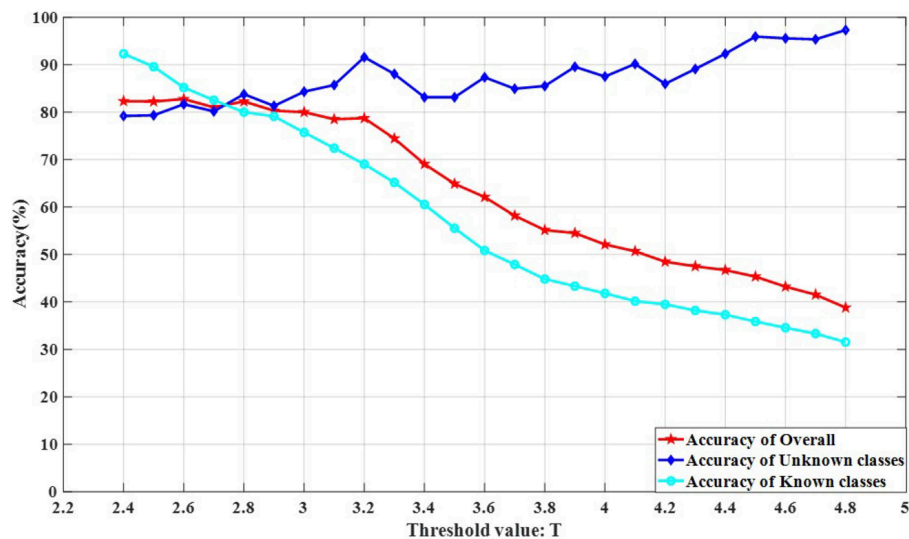
FIGURE 12 | The influence of OSKKNN threshold value *T* on the recognition accuracy of rotate 180° horizontally data.

of unknown classes appear in the known classes test data, the more difficult it is to recognize the samples of known classes. As the threshold value *T* increases, the recognition effect of the unknown classes is better. This is because the larger the threshold value *T* is, the fewer the samples of known classes are in the unknown classes. The recognition accuracy is no longer affected by the sample classes, but only depends on the performance of the classifier 2 (Kernel-KNN). The overall recognition accuracy rate has an inconspicuous upward trend, followed by a significant downward trend. This is because the rising stage, the recognition accuracy rate of known classes decreases slightly, while the recognition accuracy rate of unknown classes has an insignificant rising trend, and the rising range is larger than the falling range

of the known classes. The obvious downward trend is because the decline rate of the recognition accuracy rate of known classes is greater than the increase of the unknown classes, and the decline rate of known classes has a greater impact on the overall recognition accuracy rate.

It can be concluded from the experiment that the threshold value *T* of the data set on rotate 180° is in the range of 3.2–3.6, and the overall recognition effect of OSKKNN is better. For the data set of rotating 180° horizontally, the threshold value *T* is in the range of 2.8–3.2, and the overall recognition effect of OSKKNN is better.

When the threshold value is set to 3.3, the classification confusion matrix of the rotate 180° dataset is obtained as shown
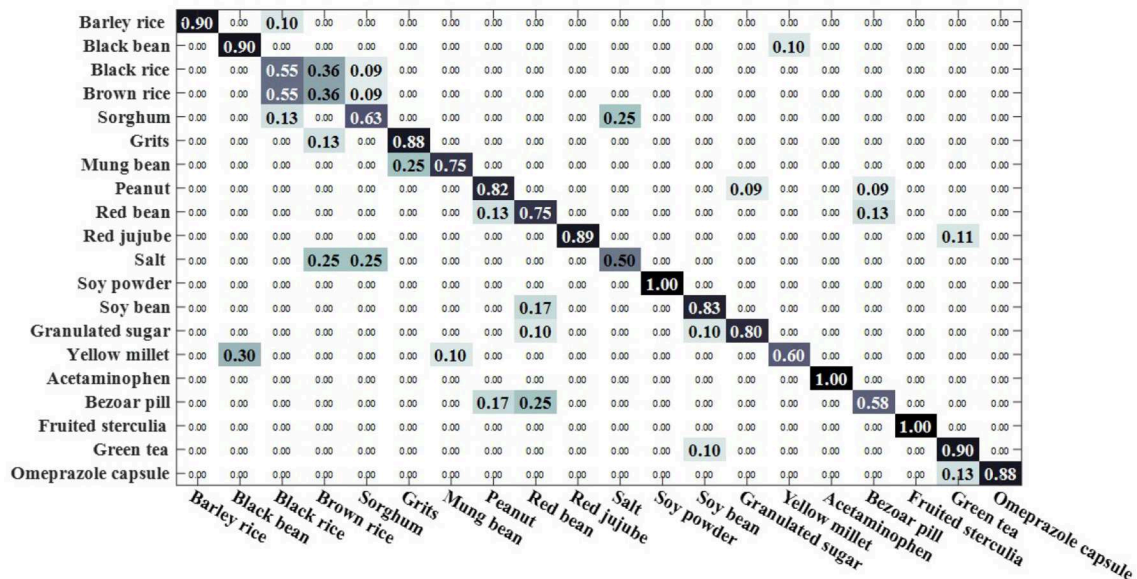
FIGURE 13 | OSKKNN recognition confusion matrix on rotate 180° data when threshold value $T = 3.3$.
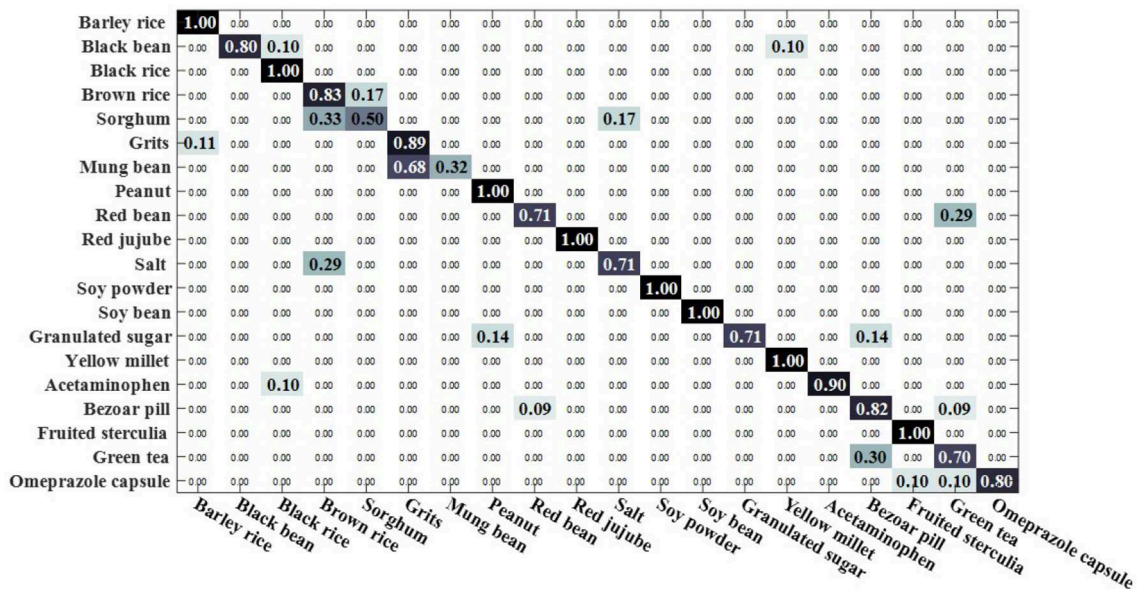
FIGURE 14 | OSKKNN recognition confusion matrix on rotate 180° horizontally data when threshold value $T = 2.9$.

in **Figure 13**, where black rice, brown rice and sorghum are hard to distinguish. 25% of salt was recognized as sorghum, 25% of salt was recognized as brown rice, 30% of yellow millet was recognized as black bean, and 25% of bazaar pill was recognized as red bean. When the threshold value is set to 2.9, the classification confusion matrix of rotate 180° horizontally can be obtained, as shown in **Figure 14**. Brown rice and sorghum are hard to distinguish. 68% of mung bean was recognized as grits, and the error rate was higher. 29%

of red bean was recognized as green tea, 29% of salt was recognized as brown rice, 14% of white sugar was recognized as peanut, 14% of white sugar was recognized as bezoar pill, and 30% of green tea was recognized as bezoar pill. The overall recognition effect is better, and the recognition effect is equivalent in a closed environment. It shows that the method proposed in this paper can solve the problem of recognizing unknown class objects in an open environment and finally recognize all class objects.
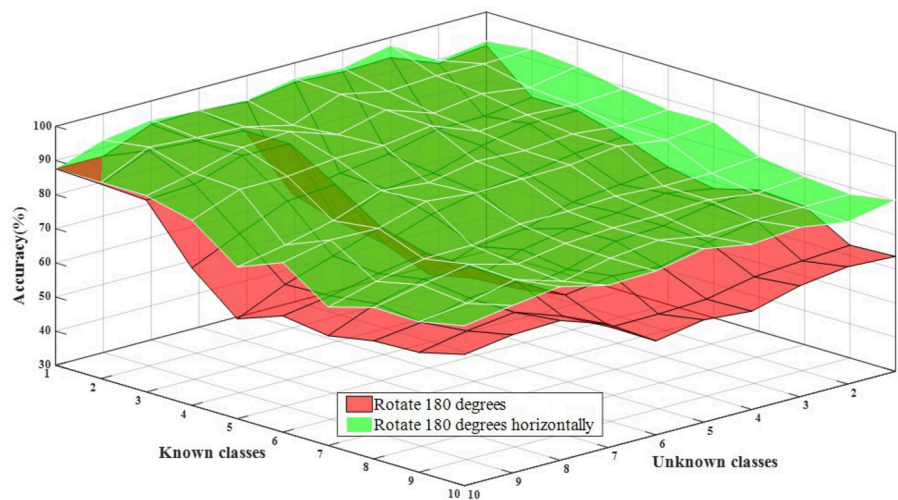
FIGURE 15 | The influence of known classes and unknown classes on the overall accuracy of recognition.

5.2.3. Performance Evaluation for Recognizing All Class Objects in Open Environment

The proposed framework mainly addresses the recognition of all class objects in an open environment, so the effectiveness of the method can be assessed using the openness to the overall recognition accuracy. Openness refers to the ratio of known classes and unknown classes, so the factors affecting the accuracy of recognition are the number of unknown classes and the number of known classes. Set 10 classes of known classes and 10 classes of unknown classes, rotate 180° data set at a threshold of 3.3 for experimentation, and rotate 180° horizontally data set at a threshold of 2.9 for experiments.

Experimental results are shown in **Figure 15**. From the test results, it can be concluded that the more known the classes, the worse the recognition effect is. In other words, the recognition accuracy decreases with the increase of the number of known classes, and the performance is stable. For the number of unknown classes, the recognition accuracy of rotating 180° horizontally fluctuates less, while that of rotating 180° fluctuates more. This is because there are some confusing categories and different interaction methods, but a relatively good recognition effect is reached. Therefore, the proposed method can solve the problem of audio recognition for all classes in an open environment.

6. CONCLUSION

With the rapid development of intelligent service robots and the increasing demand, intelligent robots can detect the objects in the container. Because of the limitations of vision and touch, the objects in the container cannot be detected. Therefore, different interaction methods are used to collect the auditory information (sound) for recognizing objects in containers. The experiment proves that the robot can recognize objects in containers well with audio, and the recognition effect is better with the single

joint rotation of 180° or horizontal rotation of 180°. It provides a good way for the intelligent service robot to recognize objects in containers when interacting with these containers, which has high application value.

In real life, people are constantly learning and exploring unknowns. Traditional machine learning is carried out in a closed environment, which does not conform to the mode of intelligent robots and people. Therefore, the Kernel-KNN algorithm is improved and extended to solve the problem of audio recognition in an open environment in this paper. Experiments show that the proposed OSKKNN algorithm has a good recognition effect and can solve the problem of using audio to recognize objects in an open environment. It also provides a feasible idea for other fields such as tactile and visual fields.

However, there is still the problem of object confusion in the recognition process, and several classes of object recognition effects are not ideal. Therefore, more economical, simple and convenient multi-modal sensors need to be developed in the future to collect information of multiple modes for better recognition. Develop the more stable, fast and low running cost algorithm. Develop more modal fusion and modal pairing algorithms (Zheng et al., 2019a,b). Combine visual and tactile matching for recognition (Liu et al., 2017, 2018a,b). In the case of reducing the noise of the robot itself, data collection through the integration of multiple interactive actions can improve the recognition accuracy. Recognizing in an open environment is a new research problem that requires more in-depth theoretical research and the development of open recognition methods suitable for all modalities.

DATA AVAILABILITY STATEMENT

The raw data supporting the conclusions of this manuscript will be made available by the authors, without undue reservation, to any qualified researcher.

AUTHOR CONTRIBUTIONS

SJ: contribution was to propose theoretical methods, establish data sets, experiments, and paper writing. HL: contribution was to instruct the theory, experiments and paper writing, and project support. BW: contribution was project support. FS: contribution was to guide paper writing.

FUNDING

REFERENCES

Araki, T., Nakamura, T., Nagai, T., Funakoshi, K., Nakano, M., and Iwahashi, N. (2011). "Autonomous acquisition of multimodal information for online object concept formation by robots," in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems* (San Francisco, CA: IEEE), 1540–1547.

Bao, J., Chen, D., Wen, F., Li, H., and Hua, G. (2018). "Towards open-set identity preserving face synthesis," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (Salt Lake City, UT).

Bapst, A. B., Tran, J., Koch, M. W., Moya, M. M., and Swahn, R. (2017). "Open set recognition of aircraft in aerial imagery using synthetic template models," in *Automatic Target Recognition XXVII* (Anaheim, CA: International Society for Optics and Photonics), 1020206.

Bendale, A., and Boult, T. E. (2016). "Towards open set deep networks," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (Las Vegas, NV).

Beran, M. J. (2012). Quantity judgments of auditory and visual stimuli by chimpanzees (pan troglodytes). *J. Exp. Psychol. Anim. Behav. Process.* 38:23. doi: 10.1037/a0024965

Berthouze, L., Prince, C., Littman, M., Kozima, H., and Balkenius, C. (2007). "Shaking eases object category acquisition: experiments with a robot arm," in *7th International Conference on Epigenetic Robotics* (Piscataway, NJ).

Boiman, O., Shechtman, E., and Irani, M. (2008). "In defense of nearest-neighbor based image classification," in *2008 IEEE Conference on Computer Vision and Pattern Recognition* (Anchorage, AK: IEEE), 1–8.

Brichetto, D., Carlson, B., Gaston, M. L., Olson, T., and Loebach, J. (2018). A comparison of free-field and headphone based sound localization tasks. *J. Acoust. Soc. Am.* 143, 1814–1814. doi: 10.1121/1.5035950

Cao, J., Zhao, T., Wang, J., Wang, R., and Chen, Y. (2017). Excavation equipment classification based on improved mfcc features and elm. *Neurocomputing* 261, 231–241. doi: 10.1016/j.neucom.2016.03.113

Chang, C.-C. and Lin, C.-J. (2011). Libsvm: a library for support vector machines. *ACM Trans. Intell. Syst. Techn.* 2:27. doi: 10.1145/1961189.1961199

Chang, X., Yu, Y.-L., Yang, Y., and Xing, E. P. (2016). Semantic pooling for complex event analysis in untrimmed videos. *IEEE Trans. Patt. Analy. Mach. Intell.* 39, 1617–1632. doi: 10.1109/TPAMI.2016.2608901

Chen, C. L., Snyder, J. O., and Ramadge, P. J. (2016). "Learning to identify container contents through tactile vibration signatures," in *2016 IEEE International Conference on Simulation, Modeling, and Programming for Autonomous Robots (SIMPAR)* (San Francisco, CA: IEEE), 43–48.

Clarke, S., Rhodes, T., Atkeson, C. G., and Kroemer, O. (2018). "Learning audio feedback for estimating amount and flow of granular material," in *Conference on Robot Learning* (Zürich) 529–550.

Do, H. M., Sheng, W., Liu, M., and Zhang, S. (2016). "Context-aware sound event recognition for home service robots," in *2016 IEEE International Conference on Automation Science and Engineering (CASE)* (Fort Worth, TX: IEEE), 739–744.

Eppe, M., Kerzel, M., Strahl, E., and Wermter, S. (2018). "Deep neural object analysis by interactive auditory exploration with a humanoid robot," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Madrid: IEEE), 284–289.

Forero, M. G., Herrera-Rivera, S., Ávila-Navarro, J., Franco, C. A., Rasmussen, J., and Nielsen, J. (2018). "Color classification methods for perennial weed detection in cereal crops," in *Iberoamerican Congress on Pattern Recognition* (Madrid: Springer), 117–123. doi: 10.1007/978-3-030-13469-3-14

Gandarias, J., Gómez-de Gabriel, J., and García-Cerezo, A. (2018). Enhancing perception with tactile object recognition in adaptive grippers for human–robot interaction. *Sensors* 18:692. doi: 10.3390/s18030692

Grassi, M. (2005). Do we hear size or sound? balls dropped on plates. *Percept. Psychophys.* 67, 274–284. doi: 10.3758/BF03206491

Griffith, S., Sukhoy, V., Wegter, T., and Stoytchev, A. (2012). "Object categorization in the sink: learning behavior–grounded object categories with water," in *Proceedings of the 2012 ICRA Workshop on Semantic Perception, Mapping and Exploration* (Saint Paul, MN: Citeseer).

Gunther, M., Cruz, S., Rudd, E. M., and Boult, T. E. (2017). "Toward open-set face recognition," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops* (Honolulu, HI).

Guru, D., Sharath, Y., and Manjunath, S. (2010). Texture features and KNN in classification of flower images. *Int. J. Comput. Appl.* 1, 21–29.

Kaljahi, M. A., Shivakumara, P., Ping, T., Jalab, H. A., Ibrahim, R. W., Blumenstein, M., et al. (2019). A geometric and fractional entropy-based method for family photo classification. *Exp. Syst. Appl. X* 3:100008. doi: 10.1016/j.eswax.2019.100008

Kibanov, M., Becker, M., Mueller, J., Atzmueller, M., Hotho, A., and Stumme, G. (2018). "Adaptive knn using expected accuracy for classification of geo-spatial data," in *Proceedings of the 33rd Annual ACM Symposium on Applied Computing* (New York, NY: ACM), 857–865.

Le Roux, J., and Vincent, E. (2012). Consistent wiener filtering for audio source separation. *IEEE Signal Process. Lett.* 20, 217–220. doi: 10.1109/LSP.2012.2225617

Li, Z., Nie, F., Chang, X., and Yang, Y. (2017). Beyond trace ratio: weighted harmonic mean of trace ratios for multiclass discriminant analysis. *IEEE Trans. Knowled. Data Eng.* 29, 2100–2110. doi: 10.1109/TKDE.2017.2728531

Liang, H., Li, S., Ma, X., Hendrich, N., Gerkmann, T., and Zhang, J. (2019). Making sense of audio vibration for liquid height estimation in robotic pouring. *arXiv preprint* arXiv:1903.00650.

Liao, T. W., and Kuo, R. (2018). Five discrete symbiotic organisms search algorithms for simultaneous optimization of feature subset and neighborhood size of knn classification models. *Appl. Soft Comput.* 64, 581–595. doi: 10.1016/j.asoc.2017.12.039

Liu, C., Sharan, L., Adelson, E. H., and Rosenholtz, R. (2010). "Exploring features in a bayesian framework for material recognition," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (San Francisco, CA: IEEE), 239–246.

Liu, H., Sun, F., Fang, B., and Lu, S. (2017). Multimodal measurements fusion for surface material categorization. *IEEE Trans. Instrument. Measure.* 67, 246–256. doi: 10.1109/TIM.2017.2764298

Liu, H., Sun, F., and Zhang, X. (2018a). Robotic material perception using active multi-modal fusion. *IEEE Trans. Indust. Elect.* 66, 9878–9886. doi: 10.1109/TIE.2018.2878157

Liu, H., Wang, F., Sun, F., and Zhang, X. (2018b). Active visual-tactile cross-modal matching. *IEEE Trans. Cogn. Dev. Syst.* 11, 176–187. doi: 10.1109/TCDS.2018.2819826

Luo, S., Mou, W., Althoefer, K., and Liu, H. (2016). "Iterative closest labeled point for tactile object shape recognition," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Daejeon: IEEE), 3137–3142.

Luo, S., Zhu, L., Althoefer, K., and Liu, H. (2017). Knock-knock: acoustic object recognition by using stacked denoising autoencoders. *Neurocomputing* 267, 18–24. doi: 10.1016/j.neucom.2017.03.014

Moeini, A., Faez, K., Moeini, H., and Safai, A. M. (2017). Open-set face recognition across look-alike faces in real-world scenarios. *Image Vision Comput.* 57, 1–14. doi: 10.1016/j.imavis.2016.11.002

Nakamura, T., Araki, T., Nagai, T., Nagasaka, S., Taniguchi, T., and Iwahashi, N. (2013). "Multimodal concept and word learning using phoneme sequences with

errors," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems* (Tokyo: IEEE), 157–162.

Nakamura, T., Nagai, T., and Iwahashi, N. (2009). "Grounding of word meanings in multimodal concepts using lda," in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems* (St. Louis, MO: IEEE), 3943–3948.

Neumann, M., Nottensteiner, K., Kossyk, I., and Marton, Z.-C. (2018). "Material classification through knocking and grasping by learning of structure-borne sound under changing acoustic conditions," in *2018 IEEE 14th International Conference on Automation Science and Engineering (CASE)* (Munich: IEEE), 1269–1275.

Patel, V. M., Nasrabadi, N. M., and Chellappa, R. (2011). Sparsity-motivated automatic target recognition. *Appl. Opt.* 50, 1425–1433. doi: 10.1364/AO.50.001425

Pillai, J. K., Patel, V. M., Chellappa, R., and Ratha, N. K. (2011). Secure and robust iris recognition using random projections and sparse representations. *IEEE Trans. Patt. Analy. Mach. Intell.* 33, 1877–1893. doi: 10.1109/TPAMI.2011.34

Scalart, P., and Filho, J. V. (1996). "Speech enhancement based on a priori signal to noise estimation," in *1996 IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings* (Atlanta, GA: IEEE), 629–632.

Schenck, C., Sinapov, J., Johnston, D., and Stoytchev, A. (2014). Which object fits best? solving matrix completion tasks with a humanoid robot. *IEEE Trans. Auton. Mental Dev.* 6, 226–240. doi: 10.1109/TAMD.2014.2325822

Sinapov, J., Bergquist, T., Schenck, C., Ohiri, U., Griffith, S., and Stoytchev, A. (2011). Interactive object recognition using proprioceptive and auditory feedback. *Int. J. Robot. Res.* 30, 1250–1262. doi: 10.1177/0278364911408368

Sinapov, J., Schenck, C., and Stoytchev, A. (2014). "Learning relational object categories using behavioral exploration and multimodal perception," in *2014 IEEE International Conference on Robotics and Automation (ICRA)* (Hong Kong: IEEE), 5691–5698.

Sinapov, J., and Stoytchev, A. (2009). "From acoustic object recognition to object categorization by a humanoid robot," in *Proceedings of the RSS 2009 Workshop-Mobile Manipulation in Human Environments* (Seattle, WA).

Sinapov, J., Wiemer, M., and Stoytchev, A. (2009). "Interactive learning of the acoustic properties of household objects," in *2009 IEEE International Conference on Robotics and Automation* (Kobe: IEEE), 2518–2524.

Strese, M., Boeck, Y., and Steinbach, E. (2017). "Content-based surface material retrieval," in *2017 IEEE World Haptics Conference (WHC)* (Munich: IEEE), 352–357.

Takamuku, S., Hosoda, K., and Asada, M. (2008). Object category acquisition by dynamic touch. *Adv. Robot.* 22, 1143–1154. doi: 10.1163/156855308X324820

Taniguchi, T., Yoshino, R., and Takano, T. (2018). Multimodal hierarchical dirichlet process-based active perception by a robot. *Front. Neurorobot.* 12:22. doi: 10.3389/fnbot.2018.00022

Weinberger, K. Q., Blitzer, J., and Saul, L. K. (2006). "Distance metric learning for large margin nearest neighbor classification," in *Advances in neural information processing systems*, eds B. Scholkopf, J. Platt, and T. Hofmann (Vancouver, BC: MIT Press), 1473–1480.

Yi, Z., Zhang, Y., and Peters, J. (2017). Bioinspired tactile sensor for surface roughness discrimination. *Sens. Actuat. A Phys.* 255, 46–53. doi: 10.1016/j.sna.2016.12.021

Yong, Z., Youwen, L., and Shixiong, X. (2009). An improved knn text classification algorithm based on clustering. *J. Comput.* 4, 230–237. doi: 10.4304/jcp.4.3.230-237

Yu, K., Ji, L., and Zhang, X. (2002). Kernel nearest-neighbor algorithm. *Neural Process. Lett.* 15, 147–156. doi: 10.1023/A:1015244902967

Zhang, H., Berg, A. C., Maire, M., and Malik, J. (2006). "Svm-knn: discriminative nearest neighbor classification for visual category recognition," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)* (New York, NY: IEEE), 2126–2136.

Zheng, W., Liu, H., Wang, B., and Sun, F. (2019a). Cross-modal surface material retrieval using discriminant adversarial learning. *IEEE Trans. Indust. Informat.* 15: 4978–4987. doi: 10.1109/TII.2019.2895602

Zheng, W., Liu, H., Wang, B., and Sun, F. (2019b). Online weakly paired similarity learning for surface material retrieval. *Indus. Robot.* 46, 396–403. doi: 10.1108/IR-09-2018-0179

Zöller, G., Wall, V., and Brock, O. (2018). "Acoustic sensing for soft pneumatic actuators," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Madrid: IEEE), 6986–6991.

Biceps Brachii Muscle Synergy and Target Reaching in a Virtual Environment

Liang He and Pierre A. Mathieu*

Department of Pharmacology and Physiology, Biomedical Engineering Institute, Université de Montréal, Montréal, QC, Canada

A muscular synergy is a theory suggesting that the central nervous system uses few commands to activate a group of muscles to produce a given movement. Here, we investigate how a muscle synergy extracted from a single muscle can be at the origin of different signals which could facilitate the control of modern upper limb myoelectric prostheses with many degrees of freedom. Five pairs of surface electrodes were positioned across the biceps of 12 normal subjects and electromyographic (EMG) signals were collected while their upper limbs were in eight different static postures. Those signals were used to move, within a virtual cube, a small red sphere toward different targets. With three muscular synergies extracted from the five EMG signals, a classifier was trained to identify which synergy pattern was associated with a given static posture. Later, when a posture was recognized, the result was a displacement of a red sphere toward a corner of a virtual cube presented on a computer screen. The axes of the cube were assigned to the shoulder, elbow and wrist joint while each of its the corners was associated with a static posture. The goal for subjects was to reach, one at a time, the four targets positioned at different locations and heights in the virtual cube with different sequences of postures. The results of 12 normal subjects indicate that with the muscular synergies of the biceps brachii, it was possible, but not easy for an untrained person, to reach a target on each trial. Thus, as a proof of concept, we show that features of the biceps muscular synergy have the potential to facilitate the control of upper limb myoelectric prostheses. To our knowledge, this has never been shown before.

Keywords: biceps brachii, muscle synergy, upper limb posture classification, target reaching, virtual cube, myoelectric prosthesis

INTRODUCTION

Important progress has recently been made in the design of multiple degrees of freedom (DoF) upper limb myoelectric prosthesis (Lenzi et al., 2016) and this has led the production of commercially available units, such as the Luke Arm (Mobius Bionics, 2017). Such advanced prostheses can be most valuable to amputees in their daily living. Multiple DoFs implies that multiple control signals have to be derived from EMG signals. To that end, Daley et al. (2012) used a linear discriminant analysis (LDA) to classify 12 different wrist and hand movements of normal subjects using eight optimally placed electrodes on the forearm. Similarly, Ameri et al. (2014) used an artificial neural network (ANN), where visual training was considered better than force training to simultaneously estimate intended movements of multiple joints. Comparing the classifiers performance,

Adewuyi et al. (2016) found for non-amputees and partial-hand amputees that LDA and ANN perform better than the quadratic discriminant analysis. Betthauser et al. (2018) developed a robust sparsity-based adaptive classification method to get a classification system which is appreciably less sensitive to signal deviations between training and testing. When they tested it on eight able-bodied and two transradial amputee subjects with eight electrodes pairs regularly spaced around the proximal forearm, it was found that their approach significantly outperformed other movement classification methods.

In addition to such approaches, the concept of muscle synergy was proposed to examine how the brain could efficiently command various muscles to produce different movements. For instances, to understand the posture balancing reaction of humans on a platform submitted to various perturbations in the horizontal plane, Torres-Oviedo and Ting (2007) used muscle synergies between 16 leg and lower-back human muscles. Muceli et al. (2010) found synergy among 12 muscles of the upper limb of eight subjects when reaching tasks were performed in the horizontal plane. To extract muscle synergies, various approaches can be used such as principal component analysis (PCA), independent component analysis (ICA) and non-negative matrix factorization (NMF). Amongst those, Tresch et al. (2006) considered that the NMF algorithm (Lee and Seung, 2001) was more physiologically relevant than the others given that a muscle can only be active at various contraction levels (positive) but never below rest (negative).

Features of muscle synergies are often used for classification purposes (Delis et al., 2013). For instance, Naik and Nguyen (2015) used NMF processed data to classify the finger gesture of two forearm muscles. Similarly, Rasool et al. (2016) used forearm muscles for real-time classification of hand open/close, wrist flexion/extension and forearm pronation/supination. Antuvan et al. (2016) used extreme learning machines and muscle synergy features to classify upper limb postures involved in elbow flexion/extension and shoulder flexion/protraction/retraction and rest posture. Muscle synergy has also been applied to upper limb muscles for proportional control related to prosthetic applications (Jiang et al., 2009; Ma et al., 2015).

As for our research it is focused on the multifunctional biceps brachii muscle which is involved in shoulder elevation, elbow flexion, and forearm supination (Landin et al., 2008; Jarrett et al., 2012). There is also anatomical evidence to support its multifunctionality: besides its division into two heads, its inner surface appears to be further divided into up to six compartments which are each innervated by a branch of the musculocutaneous nerve (Segal, 1992)[1]. Multifunctionality is also supported by physiological evidence: ter Haar Romeny et al. (1984) found that during different functional tasks of the upper limb, motor units of the biceps were activated at different locations within the muscle, probably due to activity in different compartments. These individual compartments can then be considered as muscles within a muscle working together to accomplish functional roles. This situation is somewhat similar to the one where anatomically different muscles work synergistically together (Bizzi and Cheung, 2013).

This paper reports on an experimental study where the biceps EMG signals are the only ones used to identify a static arm posture, out of five or eight. The study examines how successive postures could be used to develop a trajectory so as to reach a specified target in a virtual environment.

MATERIALS AND METHODS

The study was approved by the ethical committee of the Faculty of Medicine at the Université de Montréal and the 12 subjects signed a written informed consent form in accordance with the Declaration of Helsinki. To participate to the project, the inclusion criteria for each subject were: to be without any known history of neuromuscular disorders, be right-handed and aged between 20 and 35 years old; additional personal information is presented in **Table 1**.

For each subject, the borders of the biceps brachii were identified by palpation and the mid-point considered separating the short head (SH) from the long head (LH). As shown in **Figure 1A**, three pairs of surface electrodes were placed across the SH and two pairs across the LH while a reference electrode was placed over the acromion. To avoid the muscle's innervation zone, the upper row of electrodes was positioned 10 mm below the middle of the biceps. Ag/AgCl disc electrodes of 10 mm in diameter (Kendall H69P) were used with a 2 cm vertical and horizontal distance between center to center distances. Acquisition of the five EMG signals was done with customized electronic circuits using a differential amplifier (AD8226, Analog Devices) with a gain of 200. The amplified signals were rectified with an op amp (TL084, Texas Instruments), high-pass (6.67 Hz) and low-pass (1,240 Hz) filtered. Following the low-pass filter, a second gain of 10

TABLE 1 | Information on our 12 subjects with their body mass index (BMI) and mid-upper arm circumference (MUAC).

Subject ID	Height (cm)	Weight (kg)	BMI	MUAC (cm)
S1	169	50	17.5	23
S2	163	48	18.1	22
S3	167	55	19.7	24
S4	157	52	21.1	25
S5	160	58	22.7	24
S6	172	80	27.0	29
S7	183	70	20.9	28
S8	180	75	23.1	30
S9	170	72	24.9	28
S10	173	77	25.7	28
S11	179	83	25.9	32
S12	183	112	33.4	37
MEAN	171.3	69.3	23.3	27.5
±SD	8.7	18.3	4.4	4.3

---

[1]Anatomically, compartments are not unique to the biceps brachii. They are also present in the deltoid, the pectoralis major and the latissimus dorsi at the shoulder level (Brown et al., 2007) which are less frequently studied than the biceps.
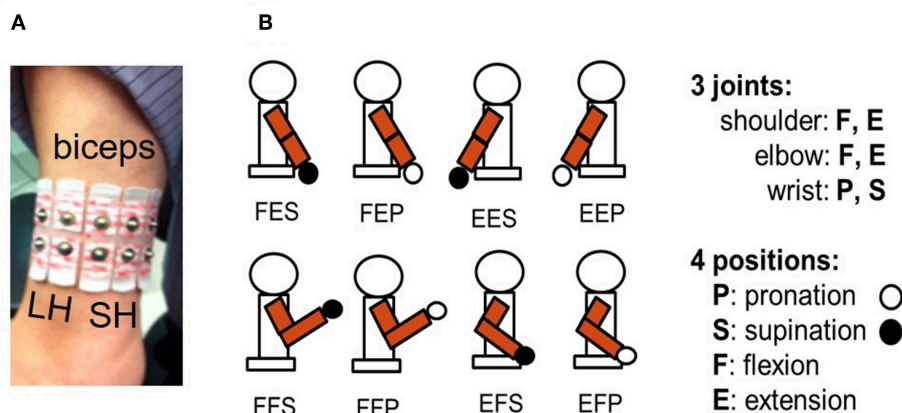
FIGURE 1 | (A) Five bipolar surface electrodes were placed across the biceps brachii: two over the long head (LH) and three across the short head (SH). **(B)** Eight different static postures of the right upper arm were used. They are identified with a 3-tuple system [*,*,*] respectively representing shoulder, elbow, and wrist joints. For the first * it can be F (flexion) or E (extension) for shoulder joint posture, second * can be F or E for elbow joint posture, and third * can be P (pronation) or S (supination) for wrist joint posture.
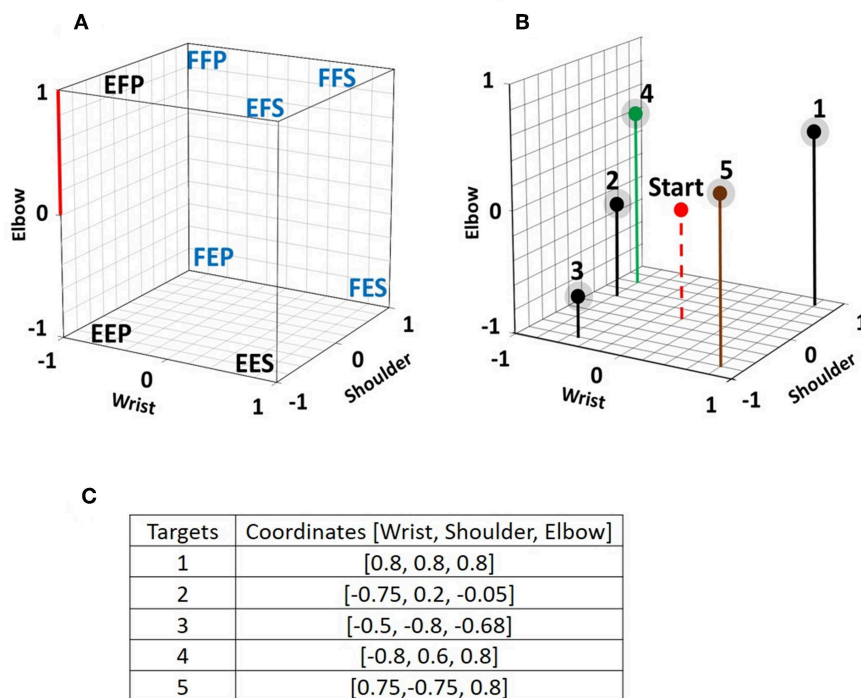
FIGURE 2 | (A) Each of the eight chosen static postures is assigned to a corner of a virtual cube displayed on a computer screen. Each axis of the cube represents the normalized angular changes at the shoulder, the elbow, and the wrist joint. Experiments were done with the eight illustrated postures and with a subset of five postures which are identified with a blue font. On the Elbow axis, the distance between 0 and 1 is used as the reference length to which distance, length and diameter measures are compared. **(B)** 3D view of the 5 targets within the cube. In each trial, the initial position of the red sphere is in the center of the cube and subjects have to move it so that it reaches one of the targets, which have a diameter of 0.2 including their grayed surrounding. The red sphere is only a point in the program, but it is displayed with a given diameter to make it visible to the subjects. **(C)** 3D coordinates of the five targets.

was obtained using the same TL084 op amp. Following this analog processing, signals were digitized (2,000 Hz, 12 bits) with a microcontroller (ROBOTIS OpenCM9.04). On an ARM Cortex-M3 processor (72 MHz clock), a root mean squared (RMS) function was implemented with a window width of

250 ms and a large 70 ms step size due to the communication rate of 15 Hz between the microcontroller and a laptop which hosted the MATLAB software that provided data processing and a graphical user interface (GUI) for interaction with the subject.
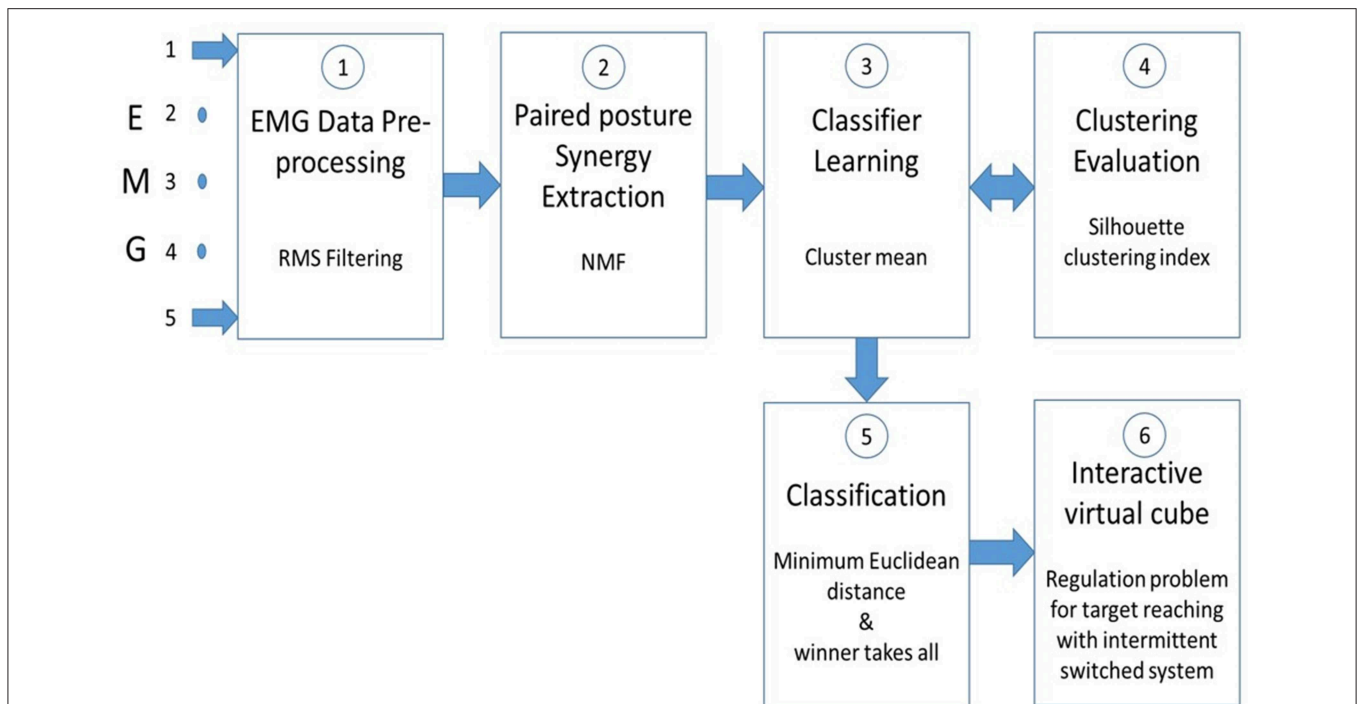
FIGURE 3 | Five raw EMG signals from the biceps are smoothed ① with a RMS filter. ② sEMG signals from paired postures are concatenated before extracting muscle synergy with a non-negative matrix factorization (NMF) program. ③ Cluster mean values of the muscle synergy from pairwise postures are used to train a classifier. ④ To improve the classification accuracy, the clusters formation is evaluated with the Silhouette clustering index. ⑤ Classification of paired postures is obtained using the minimum Euclidean distance between them. In stage ⑤ a "winner takes all" method is used to make simultaneous posture classifications. In ⑥, the virtual cube, a switched system is used to move the red sphere toward a target within the cube.

While seated, subjects had to take one of eight different static postures (**Figure 1B**) while facing a computer screen where a cube was displayed. As shown in **Figure 2A**, each corner of the cube was assigned to one of the eight static postures (SP). The normalized axes of the cube were defined so the [−1 to 1] range represented the full excursion of the shoulder and elbow joints which were either extended or flexed and to the wrist joint which had the hand set either in pronation or in supination. Each intermediate position in the three axes is interpolated linearly. The distance measured between 0 and 1 on the vertical elbow axis is used as the unit against which each trajectory length and distance is measured within the cube. Before each trial (**Figure 2B**) the red sphere was positioned at the center of the cube and it has to be moved toward one of the targets and touch its grayish sphere (diameter: 0.2) within 120 s to be a success otherwise the trial is a failure. The coordinates of each target are shown in **Figure 2C**. Each subject made three trials to reach a target.

Figure 3 presents a flow chart of the MATLAB program used from EMG pre-processing up to displaying the position of the red sphere within the virtual cube. An NMF algorithm was used to extract muscle synergies from pairwise postures as done previously (He and Mathieu, 2018); details of the method are presented in **Appendix 1**, **2**. The muscle synergy is extracted from concatenated EMG signals of two different postures and since no labeling information of the data is required, when a muscle

synergy is extracted from concatenated EMG, the classifier should have the power to detect a difference between each paired posture. This power is determined by a signal-to-noise ratio (SNR) where the signal is the difference between paired postures and noise is the dissimilarity of the clusters associated to each posture in the pair. The silhouette index of Rousseeuw (1987) was used to measure the discrimination power of the muscle synergy (i.e., how easy to identify different clusters). Some details of the silhouette index are also presented in **Appendix 1**. In the absence of a unique solution, the NMF algorithm was applied many times ($n = 30$) on the same pairwise posture of EMG data to find the best solution, as shown in **Appendix 2**.

For an online classification of the eight SPs of the upper limb, binary classifiers (Fürnkranz, 2002) were used with a round robin method (Park and Fürnkranz, 2007) which transforms binary classifiers into a posture classifier. The number of postures to be classified is a parameter of the classifier which determines the number of binary classifiers. With eight static postures, the number of pairwise posture classifiers needed to obtain a posture classification is $8 \times (8–1)/2 = 28$. The governing equation in the binary classifier is the measured Euclidean distance between the tested pattern and the learned class reference (equations A1.3, A1.4 in **Appendix 1**). In **Figure 4**, the 21 thin lines connecting a pair of eight postures represent a trained binary classifier and the seven thick lines related to the round robin method are used to identify a posture such as FFS. For the five SP condition (a
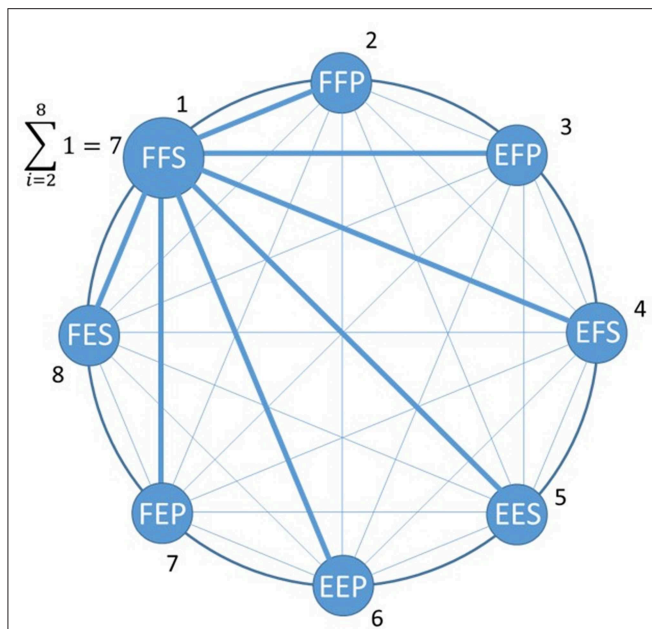
FIGURE 4 | Posture classification example where the maximum number of paired posture classification is 8*(8−1)/2 = 28. Out of those there are 21 thin blue lines representing paired posture classifier associated with postures other than FFS which were taken by a subject. Associated with FFS are seven paired classifiers (large blue lines) which make correct classification with FFS.

subset of the eight SPs), 10 pairwise posture classifiers are used. The posture identified with the posture classifier is fed to an intermittent controller.

Intermittent Controller

For each trial, the initial location of the red sphere is at the center of the cube and when a first static posture taken by the subject to reach a target is identified by the classifier (**Figure 5**), the intermittent controller (Gawthrop et al., 2011) moves the red sphere toward the corner of the cube associated to the detected posture. If the target is not reached, additional posture changes are produced up to when the target is reached or when a 120 s time limit expires. Within the intermittent controller a discrete state switch control is used to compare the new joint posture with the previous one. Then, the activated joints are only those where a change had occurred. For example, in a FFS to FEP posture change, the shoulder joint (first F in both postures) will be inactivated and the red sphere will move, from flexion (F) to extension (E) along the elbow axis, and simultaneously on the wrist axis, from supination (S) to the pronation (P). When a change occurs simultaneously at the three joints as from EFS to FEP, the shoulder joint will only be activated and the red sphere will move, along the shoulder axis from extension (E) to flexion (F). As for the two other joints, they will remain inactive until the subject makes another posture change which does not involve the shoulder joint.
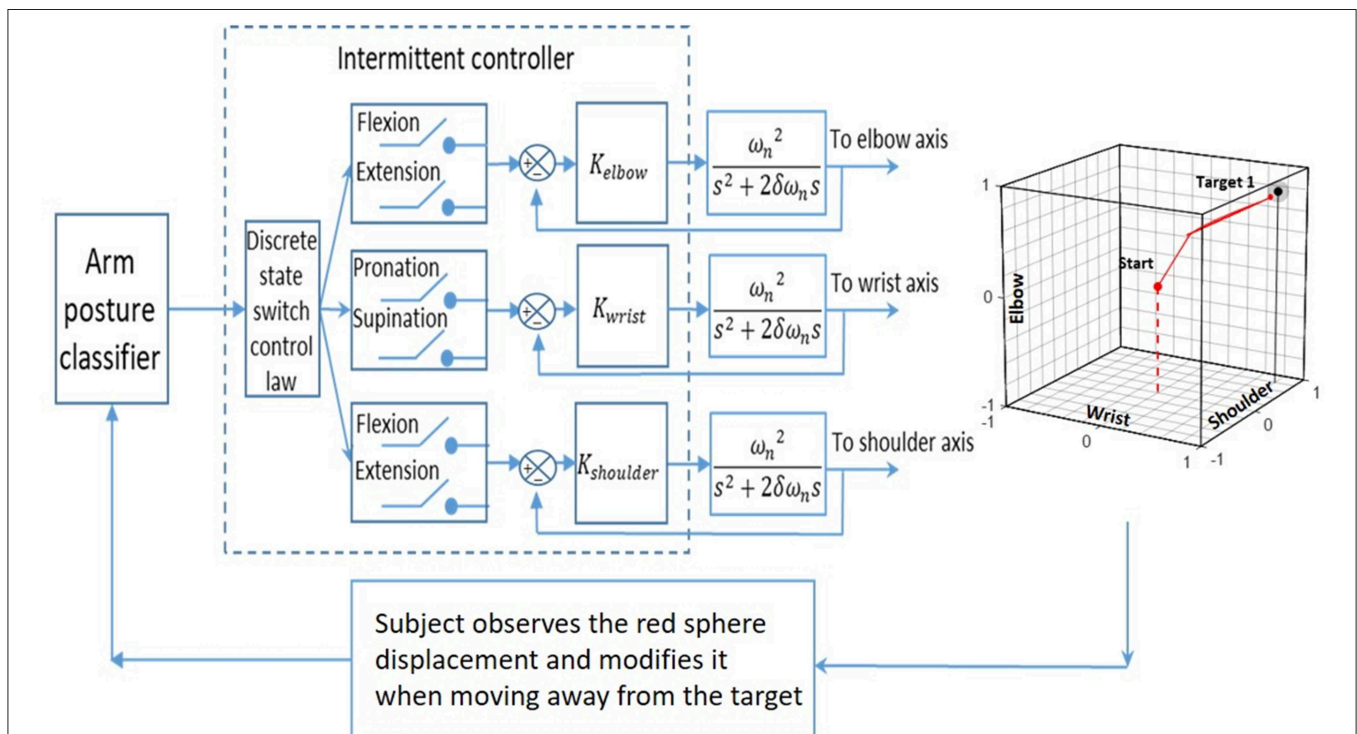
FIGURE 5 | The input of an arm posture classifier is fed to an intermittent controller that determines which switch to close in order to move the red sphere accordingly with the subject's posture. The movement of the red sphere in each of the three joint axes is produced with a 2nd order system whose transfer function is given by Equation 1. The subject used the movement of the red sphere within the cube as visual feedback to produce sequential posture changes, and the objective was to reach a target in the cube.

Red Sphere Displacement

The position of the red sphere in the cube corresponds to the position of the three joints of the upper limb of a subject and a static posture change is associated with a step function input to a second-order system and the output is a continuous displacement of the red sphere toward the appropriate corner. For this second-order system (Equation 1), the resistance to the movement of the red sphere is associated with ω_n (a large value produces a larger resisting force to the displacement) and resistance to speed change of the red sphere is associated with δ (a larger value produces a larger resisting force). The transfer function of the second-order system is:

$$\frac{C(s)}{R(s)} = \frac{\omega_n^2}{s^2 + 2\delta\omega_n s + \omega_n^2} \qquad (1)$$

where $C(s)$ is the output to the red sphere and $R(s)$ is a step function associated with each static posture change made by the subject. The dynamic parameters of the red sphere's movement ($\omega_n = 0.39$ and $\delta = 1$) were the same for each subject. When a new posture is identified while the red sphere is still moving; the red sphere immediately changed its direction under the actuation of the intermittent controller. To move the red sphere anywhere within the cube, at least four static postures are needed.

Protocol

The day before the experiment, each subject came to the laboratory to view a short video demonstrating how to produce each of the eight static postures illustrated in **Figure 1B** and during ~150 min, they practiced controlling the movements of the red sphere. The next day, during the experiment, targets 1, 2, 3, 4 had to be reached in succession with five SPs and targets 1, 2, 3, 5 with eight SPs. Targets 1, 2, 3 were thus considered with both the five and eight SP groups for comparison purposes. Performance of target reaching was measured with four measures: (1) time in seconds to reach a target; (2) trajectory length made by the red sphere from its initial position up to reaching a target, or in a failed trial, up to its end position when 120 s was reached; (3) number of posture changes taken to reach a target or to reach the 120 s for a failed trial; (4) the distance between the red sphere and the target when 120 s was reached for a failed trial. The Runge-Kutta 4 (RK4) method is used to obtain the trajectory length of the red sphere, which is obtained from the cumulative sum of Euclidean distances along the numerical solutions.

During the classifier training and evaluation, subjects kept their arm in each of the eight static postures for 10 s, from which the muscular synergies were extracted for classifier training. Next, they kept three times each of the five or eight static posture for 5 s, from which the obtained synergies were compared online with the trained classifier. For each trial, 30 iterations were produced by the classifier and a percentage of good classification was obtained and a mean value obtained for each subject for the five and eight SPs condition.

The NASA task load index (TLX) survey form (**Appendix 3**) was filled by each subject after the experiment. This is a self-evaluation of six items: mental, physical, temporal demand, and level of effort, frustration, and performance during the experiment. That feedback could provide valuable information to improve the experimental protocol and software programs used to process the information.

Statistics

A paired t-test was used to compare mean results obtained by the 12 subjects when five or eight SPs were used. The tested hypothesis was that with five SPs, the mean classification results and the number of targets reached would be better than with eight SPs because remembering how to reproduce with some fidelity eight different postures is more mentally demanding than for only five. Where numbers of subjects were different for targets or postures (**Figures 7**, **8**), independent sample t-test were used to test the difference between the compared results. A difference was considered significant when $p < 0.05$ and the IBM SPSS Statistics software was used.

RESULTS

Classification % obtained during the training with five and eight SPs are shown in **Table 2A**. With five SPs, the classification of four subjects was very good (>90%), although it was quite poor for S12 (31%). With eight SPs, the mean classification value (of 72 ± 20%) was significantly lower ($p = 0.001$) than with five SPs (82 ± 19 %). In **Table 2B**, each subject's ratio of the number of reached targets out of 12 (3 trials × 4 targets) is presented. It can be observed that a good classification % in the training session was not always associated with a large ratio of reached targets. For instance, S2 and S8, who were among the five subjects with high classification performance, did not reach a single target in the eight SP condition. As expected however, S12 with the lowest classification results could only reach one target with five SPs and 0 with eight SPs (ratio: 0.08 and 0.00). For the group, the mean ratio of target reached was higher with five SPs (0.29 ± 0.18) than with eight SPs (0.24 ± 0.23) but this difference was not significant ($p = 0.281$).

Figure 6 illustrates two trajectories of the red sphere which was controlled by S3 trying to reach target two with five SPs. In **Figure 6A**, an example of a failed trial is shown, where in spite of 72 posture changes made during 120 s, the red sphere was still at a distance of 0.68^2 from target two after a trajectory length of 14.7^2. The same subject was far more successful in another trial (**Figure 6B**) where the same target was reached with only four posture changes within 9 s.

The mean time to successfully reach the targets is shown in **Figure 7A**. Target one was the easiest to reach with a mean time of 38 and 20 s for the five SP and eight SP condition, respectively, and target three was the most difficult to reach with 59 and 55 s. These mean values are the results of significant variations among the subjects and no significant difference was found between those results. Reaching target one was achieved with a smaller number of posture changes

---

[2] Length is referenced to the 0 to 1 distance on the elbow axis (**Figure 2A**).

TABLE 2 | (A) In the training session, mean (± SD) classifier accuracy (%) for the 5 and 8 static postures (SP) conditions.

Subjects	5 SP	8 SP	Subjects	5 SP	8 SP
	Mean ± SD	Mean ± SD		Ratio	Ratio
A			**B**		
S5	99 ± 2	87 ± 20	S3	0.67	0.75
S11	95 ± 4	68 ± 35	S8	0.50	0.00
S1	94 ± 6	88 ± 25	S7	0.50	0.25
S2	90 ± 10	83 ± 20	S1	0.42	0.58
S8	89 ± 9	79 ± 21	S6	0.33	0.33
S3	87 ± 14	75 ± 43	S5	0.25	0.25
S4	87 ± 21	90 ± 11	S9	0.25	0.25
S7	87 ± 15	85 ± 13	S11	0.17	0.17
S9	85 ± 15	72 ± 26	S4	0.17	0.17
S10	76 ± 16	64 ± 26	S2	0.08	0.00
S6	60 ± 36	49 ± 45	S10	0.08	0.08
S12	31 ± 45	22 ± 42	S12	0.08	0.00
Mean ± SD	82 ± 19	72 ± 20	Mean ± SD	0.29 ± 0.18	0.24 ± 0.23

Mean values of each subject are obtained from three trials of 5 s in each of the five or in each of the eight static postures. The subjects are sorted from highest to lowest performance in the 5 SP condition. The difference between 5 and 8 SP results is significant ($p < 0.05$). (B) Success ratio [number of successful trials over 12 trials (4 targets × 3 times)] by each subject. The subjects are sorted from highest to lowers performance in the 5 SP condition. S3 was the best performer. Between the 5 and 8 SP results the paired t-test value is 0.28.

FIGURE 6 | Trajectories taken by S3 to move the red sphere toward target 2 in the 5 SP condition. (A) An unsuccessful 1st trial in spite of 72 posture changes where the final red sphere position after 120 s was at a distance of 0.68 (small black circle) from the target. (B) In the 3rd trial, the target was reached within 9 s with only 4 posture changes.
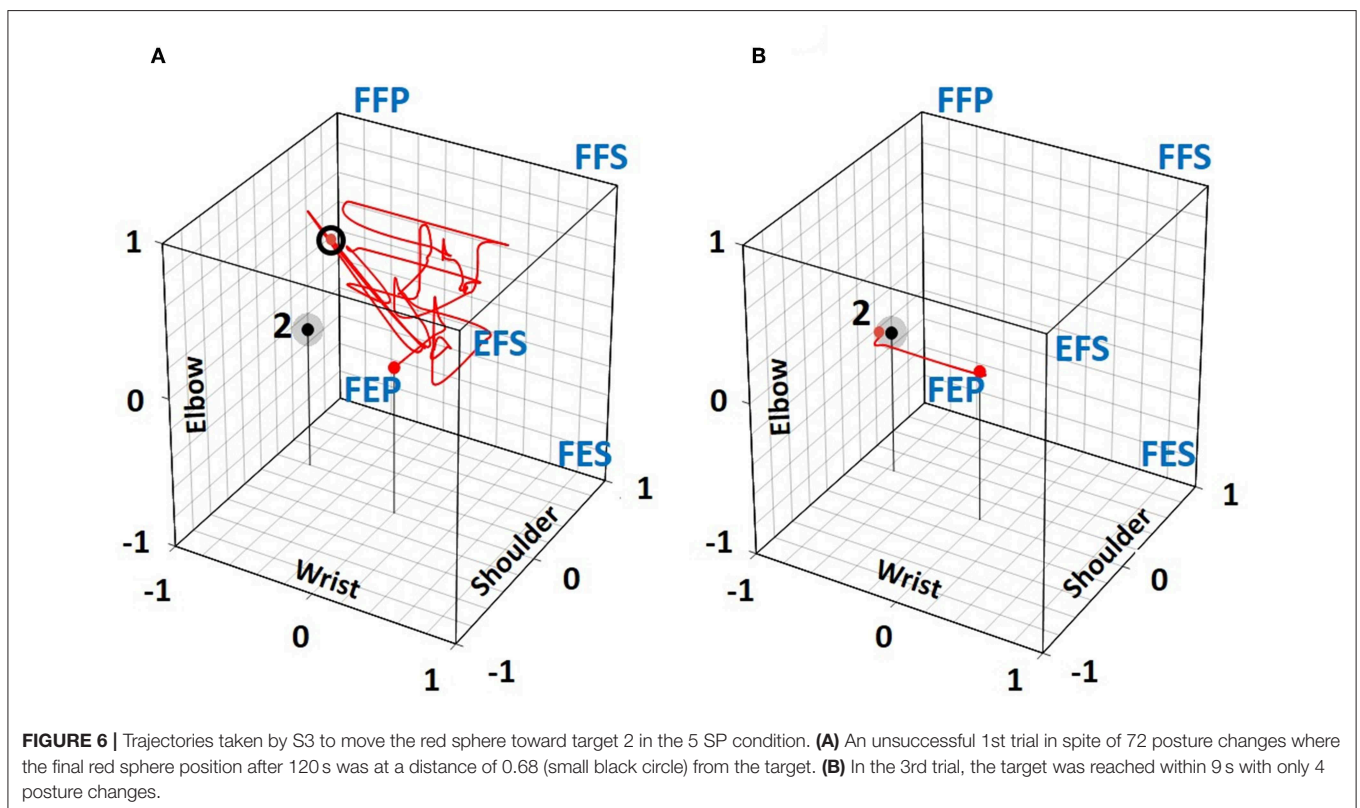
(**Figure 7B**) than for the other targets. With eight SPs, a significant difference was found between targets one and two and between one and five. For trajectory length (**Figure 7C**), a significant difference was obtained with eight SPs between targets one and three and with five SPs between targets two and three.

Results for failed trials are shown in **Figure 8**. In **Figure 8A**, it is seen that mean distances separating the red sphere from target two at the end of 120 s are smaller than those for the other targets. For the five SP condition the differences are significant between targets 1 2, 3, and 4 while for the eight SP condition the difference is only significant between targets two and three. While
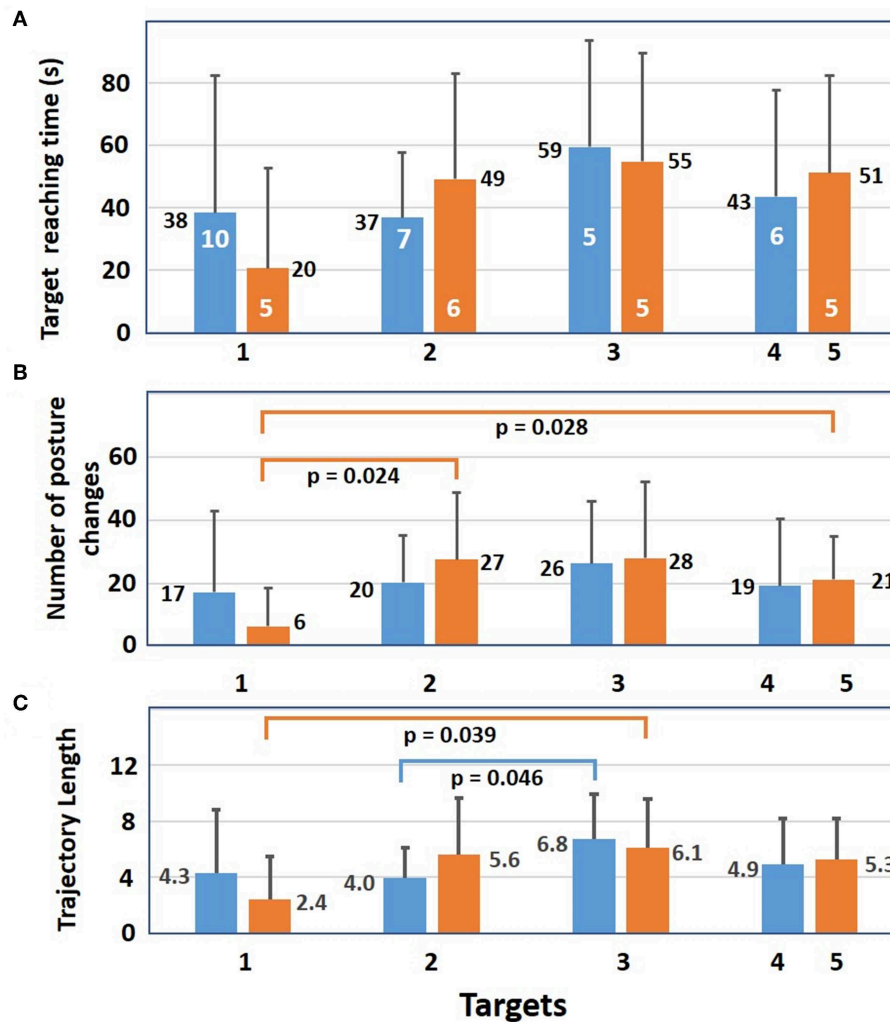
FIGURE 7 | For successful trials, mean results (+sd) of the 12 subjects with 5 (blue bars) and with 8 static postures (orange bars). **(A)** Mean time needed to reach each target. Number of subjects who reached a target at least once is shown in white. **(B)** Mean number of posture changes needed to reach each target. For 8 static postures, there is a significant difference in the number of posture changes between targets 1 and 2 as well as between target 1 and 5. **(C)** Mean trajectory length needed to reach each target. Differences between targets 1 and 3 are significant for 8 static postures as well as targets 2 and 3 for five postures. Significance of independent samples t-test level is $p \leq 0.05$.

mean distance varied between 0.8 and 1.3, the smallest distance (0.1) was found with five SPs for target two and with eight SPs for target one and the largest was 0.8 (white number) for the eight SPs condition at target three. As for the number of mean posture changes (**Figure 8B**), they were always smaller with five SPs than with eight SPs and the only significant difference was for five and eight SPs at target three. The mean trajectory lengths of the red sphere (**Figure 8C**) were all equally elevated for five and eight SPs.

The results of the NASA task load form are presented in **Table 3**. As a group, subjects considered that physical and mental demands to identify which static posture to choose to move the red sphere were high (15.3/20 and 14.0/20). As for the temporal demand, the limit of 120 s appeared adequate and the time spend

in the lab not too long (10.4/20). In general, subjects were not very satisfied with their performance (7.2 ± 3.7) and somewhat frustrated (12.8 ± 4.9) from not having reached the targets more often. At the individual level, the best performer (S3) ranked both the physical and mental demands at a high 17/20 and frustration at the highest score of 20/20, for not being able to reproduce the postures correctly in order to reach more targets. While S2 got over 80% for the training classifier accuracy results (**Table 2A**), this subject could only reach one target (**Table 2B**). This subject considered the performance quite low (4/20) and was very frustrated (16/20). While the other low performer (S12) missed most of the targets, this subject considered, unexpectedly, that the experiment was not very mentally demanding and was not frustrating.
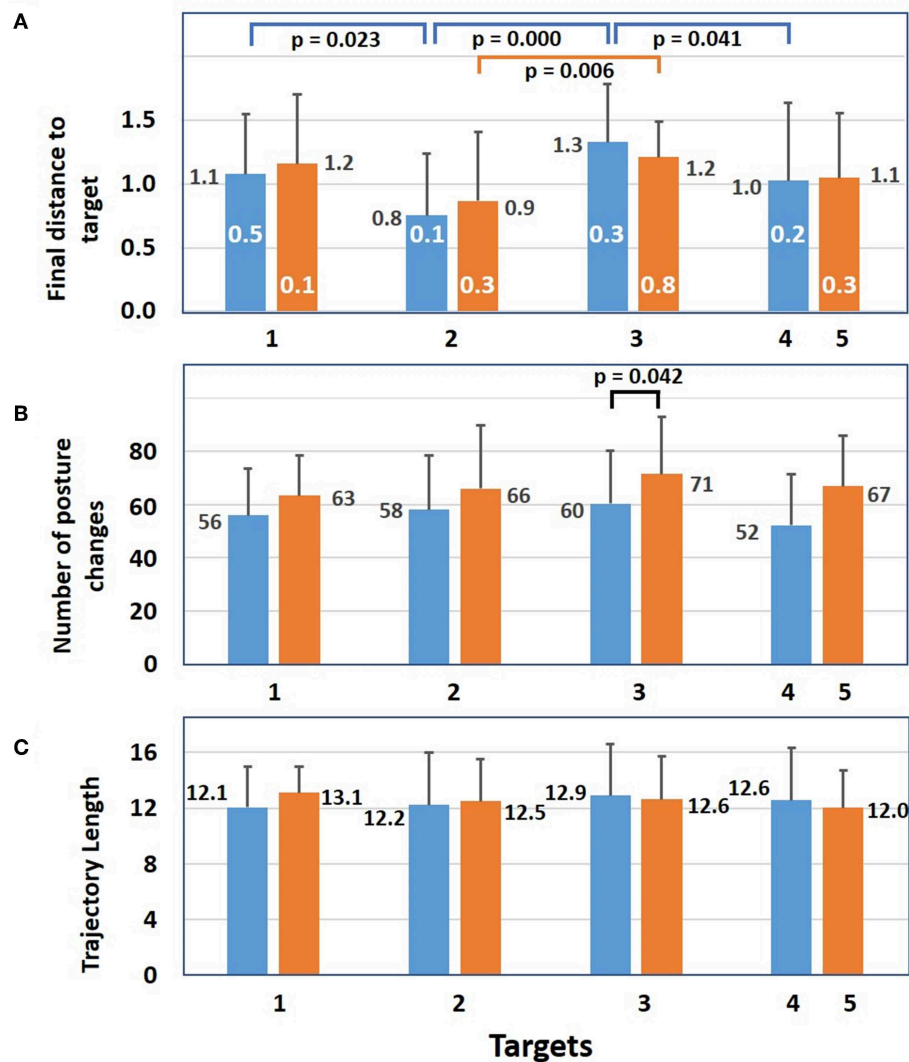
FIGURE 8 | Results of failed trials. **(A)** Mean distance (+sd) between the red sphere and a target at 120 s for 5 (blue bars) and for eight static postures (orange bars). White numbers represent the minimum distance between the red sphere and a target at 120 s. For the 5 SP condition, differences between targets 1 and 2, targets 2 and 3, and targets 3 and 4 are significant. For the 8 SP condition, the difference between targets 2 and 3 is significant. **(B)** Mean number of posture changes at the end of each trial. For target 3 the difference between 5 and 8 SP conditions is significant. **(C)** For each target, the mean trajectory length of the red sphere during 120 s (independent samples t-test significance is $p \leq 0.05$).

DISCUSSION

Modern upper limb myoelectric prostheses are now able to produce many different movements. However, following an amputation, the number of available muscles to control them is reduced and strategies to alleviate that shortage have to be developed. When the biceps brachii is still functional, one strategy could be the extraction of more than one control signal from that muscle. To investigate that possibility with non-amputee subjects, five pairs of surface electrodes were put across this multifunctional muscle. With two postures for each shoulder, elbow and wrist joints eight different static postures in the sagittal plane were used to control the displacement of a cursor toward different targets placed within a virtual cube. Results were

obtained in two experimental conditions: one with five out of the eight static postures and one with all eight static postures.

To associate a red sphere direction to each of the eight static postures, a training phase was used. After having done that with each subject, we verified the ability of a classifier to correctly recognize each of the eight SPs. As shown in **Table 2A**, an important difference is observed between results of S5 at the top of the table and S12 at its bottom. Ability to remember how to reproduce with high fidelity five or eight different static arm postures was thus quite variable among our subjects. With, a group mean value with five SPs being significantly larger than with eight SPs ($p = 0.001$), this confirms that remembering how to duplicate five postures is significantly easier than duplicating eight SPs.

TABLE 3 | NASA task load index **(TLX)** rates six factors (left column of the table) that are used to assess the subjective workload associated with the experiment.

TLX	Mean ± SD (*n* = 12)	Best performer (S3)	Lowest performer 1 (S2)	Lowest performer 2 (S12)
Physical demand	15.3 ± 2.8	17	14	13
Mental demand	14.0 ± 3.6	17	15	9
Temporal demand	10.4 ± 6.0	11	14	13
Own performance	7.2 ± 3.7	15	4	5
Own frustration	12.8 ± 4.9	20	16	8
Own effort	12.4 ± 3.4	15	13	11

The three top factors are related to constraints associated with the tasks to be realized (demand) and the three bottom factors are related to the feelings associated with those tasks (own). Each factor is scaled from 0 to 20: higher values represents higher demand or higher performance. Mean values of the 12 subjects are shown in the second column of the table. Individual results are shown for the best performer and for the two lowest performing subjects.

Since the classifier accuracy of the above results were obtained online, no data was available for an offline cross-validation. However, the classifier training was with unsupervised synergies and their features formed non-overlapping clusters. Thus, the discrimination capacity of the learned linear classifier is not an important concern since the receiver operating characteristic curve (ROC) of those features always occupies the upper half triangle. However, it is still possible that when a subject chooses a static posture, a misclassification occurs, causing the cursor to move in an unexpected direction, which is confusing for the subject. To prevent such situations, the use of a sparse representation of the classified postures as proposed by Betthauser et al. (2018) has to be added to our programs.

From the starting position of the red sphere (**Figure 2B**), targets one and five (which are located near a corner of the cube) could easily be reached by taking the FFS or EFS posture, respectively because the initial posture that starts the movement of the red sphere actives movement in all three joint axes toward the corner corresponding to that posture. Those easy reaching strategies were used by only few subjects. Targets located at a distance from a corner were more difficult to reach since different postures had to be sequentially taken to reach them.

Among the eight SPs, different subsets of five postures could have been chosen. The present choice was based on the main contribution of the biceps to the elbow flexion and forearm supination. As for the low mean target reaching ratios (<0.30, **Table 2B**) and especially for subjects S2 and S12 who reached only one target over 12 trials with five SPs and 0 targets with eight SPs, classification accuracy could be improved by replacing the present classifier by a support vector machine or an artificial neural network classifier. Also, the short training period the day before the experiment could be replaced by more training sessions as illustrated by one person of the lab who, having repeated the protocol four times, reached a 90% success rate with five SPs and 58% with eight SPs (unpublished results).

Classification results could also be improved by the addition of anatomical information on the biceps when the upper limb is in different postures. With an ultrasound probe placed at the biceps level where recording electrodes had been previously placed, changes in its shape and displacements relative to the skin surface were observed (unpublished results). In the future, with ultrasound images obtained before an experiment, position of the electrodes over the biceps could be optimized.

From the results of the NASA task load index, physical and mental demands have the highest scores indicating that our present approach to reach targets is not very intuitive. The difficulty of controlling their prostheses frequently leads amputees to leave them in a closet. Thus, it is suggested to replace the step by step cursor control of a red sphere in a virtual cube by a more realistic situation where the biceps' synergy would control a small humanoid robot for reaching objects within an arm's length. This would be a more realistic situation to the one shown in a video where the experimented person of the lab controlled a small humanoid robot arm with muscular synergies of the biceps (http://www.igb.umontreal.ca/).

CONCLUSION

We present a proof of concept that the muscular synergy extracted from a single muscle, the biceps brachii, could facilitate the control of an upper limb prosthesis. This was demonstrated by collecting five surface EMG signals of the biceps of 12 normal subjects who put their arm in eight different static postures. Using a non-negative matrix factorization program, three muscular synergies were extracted and following a training session, a classifier could identify each of those eight postures. Then, within a virtual cube displayed on a screen, subjects could, with five and eight different static postures, move a red sphere toward the targets. The number of targets reached was higher with five choices of posture than with eight choices. The reasons for a low mean number of reached targets (around 30%) were a lack of training of our subjects before the experiment and a classifier that was not lenient enough. While the biceps may not be available for above elbow amputees, a muscular synergy may then be extracted from muscles near the shoulder such as the deltoid, the pectoralis major and the latissimus dorsi, which are also multifunctional.

DATA AVAILABILITY STATEMENT

The datasets generated for this study are available on request to the corresponding author.

ETHICS STATEMENT

The studies involving human participants were reviewed and approved by Comité d'éthique de la recherche en santé, Université de Montréal. The patients/participants provided their written informed consent to participate in this study.

AUTHOR CONTRIBUTIONS

LH developed the interactive control system for target reaching, conducted the data acquisition, compiled the results, and wrote the initial draft of the article. PM designed the evaluation protocol, supervised the study, revised the article, and obtained the funding. Each author approved the final version of the manuscript.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fnbot.2019.00100/full#supplementary-material

REFERENCES

Adewuyi, A. A., Hargrove, L. J., and Kuiken, T. A. (2016). Evaluating EMG feature and classifier selection for application to partial-hand prosthesis control. *Front. Neurorobot.* 10:15. doi: 10.3389/fnbot.2016.00015

Ameri, A., Kamavuako, E. N., Scheme, E. J., Englehart, K. B., and Parker, P. A. (2014). Real-time, simultaneous myoelectric control using visual target-based training paradigm. *Biomed. Signal. Process.* 13, 8–14. doi: 10.1016/j.bspc.2014.03.006

Antuvan, C. W., Bisio, F., Marini, F., Yen, S.-C., Cambria, E., and Masia, L. (2016). Role of muscle synergies in real-time classification of upper limb motions using extreme learning machines. *J. Neuroeng. Rehabil.* 13, 1–15. doi: 10.1186/s12984-016-0183-0

Betthauser, J. L., Hunt, C. L., Osborn, L. E., Masters, M. R., Lévay, G., Kaliki, R. R., et al. (2018). Limb position tolerant pattern recognition for myoelectric prosthesis control with adaptive sparse repreentations from extreme learning, *IEEE Trans. Biomed. Eng.* 65, 770–778. doi: 10.1109/TBME.2017.2719400

Bizzi, E., and Cheung, V. C. (2013). The neural origin of muscle synergies. *Front. Comput. Neurosc.* 7:51. doi: 10.3389/fncom.2013.00051

Brown, J. M., Wickham, J. B., McAndrew, D. J., and Huang, X.-F. (2007). Muscles within muscles: coordination of 19 muscle segments within three shoulder muscles during isometric motor tasks. *J. Electromyogr. Kines.* 17, 57–73. doi: 10.1016/j.jelekin.2005.10.007

Daley, H., Englehart, K., Hargrove, L., and Kuruganti, U. (2012). High density electromyography data of normally limbed and transradial amputee subjects for multifunction prosthetic control. *J. Electromyogr. Kines.* 22, 478–484. doi: 10.1016/j.jelekin.2011.12.012

Delis, I., Berret, B., Pozzo, T., and Panzeri, S. (2013). Quantitative evaluation of muscle synergy models: a single-trial task decoding approach. *Front. Comput. Neurosc.* 7:8. doi: 10.3389/fncom.2013.00008

Fürnkranz, J. (2002). "Pairwise classification as an ensemble technique," in *European Conference on Machine Learning* (Helsinki), 97−110. doi: 10.1007/3-540-36755-1_9

Gawthrop, P., Loram, I., Lakie, M., and Gollee, H. (2011). Intermittent control: a computational theory of human control. *Biol. Cybern.* 104, 31–51. doi: 10.1007/s00422-010-0416-4

He, L., and Mathieu, P. A. (2018). Static hand posture classification based on the biceps brachii muscle synergy features. *J. Electromyogr. Kines.* 43, 201–208. doi: 10.1016/j.jelekin.2018.10.003

Jarrett, C. D., Weir, D. M., Stuffmann, E. S., Jain, S., Miller, M. C., and Schmidt, C. C. (2012). Anatomic and biomechanical analysis of the short and long head component of the distal biceps tendon. *J. Shoulder Elb. Surg.* 21, 942–948. doi: 10.1016/j.jse.2011.04.030

Jiang, N., Englehart, K. B., and Parker, P. A. (2009). Extracting simultaneous and proportional neural control information for multiple-DoF prostheses from the surface electromyographic signal. *IEEE Trans. Biomed. Eng.* 56, 1070–1080. doi: 10.1109/TBME.2008.2007967

Landin, D., Myers, J., Thompson, M., Castle, R., and Porter, J. (2008). The role of the biceps brachii in shoulder elevation. *J. Electromyogr. Kines.* 18, 270–275. doi: 10.1016/j.jelekin.2006.09.012

Lee, D. D., and Seung, H. S. (2001). "Algorithms for non-negative matrix factorization," in *Advances in Neural Information Processing Systems*, eds T. K. Leen, T. G. Dietterich, and V. Tresp (Cambridge, MA: MIT Press), 556–562.

Lenzi, T., Lipsey, J., and Sensinger, J. W. (2016). The RIC arm - a small, anthropomorphic transhumeral prosthesis. *IEEE/ASME T Mech.* 21, 2660–2671. doi: 10.1109/TMECH.2016.2596104

Ma, J., Thakor, N. V., and Matsuno, F. (2015). Hand and wrist movement control of myoelectric prosthesis based on synergy. *IEEE Transactions on Human-Machine Systems* 45, 74–83. doi: 10.1109/THMS.2014.2358634

Mobius Bionics. (2017). *Luke Arm System*. Available online at: www.mobiusbionics.com/wp-content/uploads/2017/08/Mobius-Bionics-LUKE-Product-Spec-Sheet.pdf (accessed March 4, 2019).

Muceli, S., Boye, A. T., d'Avella, A., and Farina, D. (2010). Identifying representative synergy matrices for describing muscular activation patterns during multidirectional reaching in the horizontal plane. *J. Neurophysiol.* 103, 1532–1542. doi: 10.1152/jn.00559.2009

Naik, G. R., and Nguyen, H. T. (2015). Nonnegative matrix factorization for the identification of EMG finger movements: evaluation using matrix analysis. *IEEE J. Biomed. Health Inform.* 19, 478–485. doi: 10.1109/JBHI.2014.2326660

Park, S. H., and Fürnkranz, J. (2007). "Efficient pairwise classification," in *European Conference on Machine Learning* (Warsaw), 658–665. doi: 10.1007/978-3-540-74958-5_65

Rasool, G., Iqbal, K., Bouaynaya, N., and White, G. (2016). Real-time task discrimination for myoelectric control employing task-specific muscle synergies. *IEEE. T. Neur. Sys. Reh. Eng.* 24, 98–108. doi: 10.1109/TNSRE.2015.2410176

Rousseeuw, P. J. (1987). Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *J. Comput. Appl. Math.* 20, 53–65. doi: 10.1016/0377-0427(87)90125-7

Segal, R. L. (1992). Neuromuscular compartments in the human biceps brachii muscle. *Neurosci. Lett.* 140, 98–102. doi: 10.1016/0304-3940(92)90691-Y

ter Haar Romeny, B. M., van der Gon, J. J., and Gielen, C. C. (1984). Relation between location of a motor unit in the human biceps brachii and its critical firing levels for different tasks. *Exp. Neurol.* 85, 631–650. doi: 10.1016/0014-4886(84)90036-0

Torres-Oviedo, G., and Ting, L. H. (2007). Muscle synergies characterizing human postural responses. *J. Neurophysiol.* 98, 2144–2156. doi: 10.1152/jn.01360.2006

Tresch, M. C., Cheung, V., and d'Avella, A. (2006). Matrix factorization algorithms for the identification of muscle synergies: evaluation on simulated and experimental data sets. *J. Neurophysiol.* 95, 2199–2212. doi: 10.1152/jn.00222.2005

Walking Human Detection Using Stereo Camera Based on Feature Classification Algorithm of Second Re-projection Error

Shuhuan Wen[1]*, Sen Wang[1], ZhiShang Zhang[1], Xuebo Zhang[2] and Dan Zhang[3]

[1] Key Lab of Industrial Computer Control Engineering of Hebei Province, Yanshan University, Qinhuangdao, China, [2] Institute of Robotics and Automatic Information System, Nankai University, Tianjin, China, [3] Department of Mechanical Engineering, York University, Toronto, ON, Canada

This paper presents a feature classification method based on vision sensor in dynamic environment. Aiming at the detected targets, a double-projection error based on orb and surf is proposed, which combines texture constraints and region constraints to achieve accurate feature classification in four different environments. For dynamic targets with different velocities, the proposed classification framework can effectively reduce the impact of large-area moving targets. The algorithm can classify static and dynamic feature objects and optimize the conversion relationship between frames only through visual sensors. The experimental results show that the proposed algorithm is superior to other algorithms in both static and dynamic environments.

Keywords: feature target, classification, dynamic environments, SLAM, re-projection error

INTRODUCTION

Simultaneous localization and mapping (SLAM) is a basic problem and research hotspot in the field of mobile robot research (Klein and Murray, 2007; Mur-Artal et al., 2015; Mur-Artal and Tardos, 2016; Saeedi et al., 2016). It is also one of the important conditions for mobile robot to realize autonomous navigation. The existing SLAM algorithm is mainly applied to static environment (Wen et al., 2015). However, the real environment is complex and changeable, such as people walking, door switching, changes of lighting, etc. They will bring unpredictable noise for modeling and positioning of the environment, so the existing SLAM algorithm is not very suitable for dynamic environment.

There are three main problems for SLAM of mobile robots in dynamic environment: SLAM solution, data association, and dynamic target processing.

For processing dynamic targets, there are some methods with good robust performance. For example, detected dynamic points and uncertain points regarded as abnormal points are discarded (Saeedi et al., 2016). But the traditional SLAM algorithm will result in a large deviation or even failure when the moving object is too large and the moving speed is too fast or too slow.

So, this paper studied detection method of the target with different speed and different size in dynamic environment. The feature targets are classified and the relationship between frame and frame is optimized only by a visual sensor. The classification method proposed in this paper can adapt all kinds of dynamic environment, and it can process for dynamic objectives with different speed and reduce the impact of moving objects with large area. The visual sensor can be used in hand or in different platforms.

The proposed algorithm in the paper can be applied to the following field: wearable equipment, intelligent mobile terminal equipment, micro-aircraft, and intelligent robots. In the paper, a novel second re-projection error algorithm based on the texture detection is proposed. The texture detection is integrated into a target classification algorithm, which can optimize the re-projection error algorithm. The main idea of the re-projection error algorithm is that the projection point should be moved away from the corresponding feature point when dynamic feature point of the previous frame is projected onto the current frame image. For static feature points, this re-projection error should be small. In addition, the texture detection proposed in this experiment mainly adopts the idea of interframe difference. According to the different attributes of static and dynamic feature targets, the continuous frame information of the environment is scanned and captured by using the visual sensor, and the difference and weighted information is used to complete the target detection. In the paper, the two algorithms are optimized and fused with the corresponding constraints, which can realize the classification of static and dynamic feature target points and the optimization of the relationship between frame and frame. In order to verify the real-time and validness of the proposed algorithm, the ORB and SURF feature algorithms is used to extract the environmental information. The experiments are done under four different dynamic environments, including near distance targets under static background, far distance targets under static background, one dynamic target under dynamic background and multiple dynamic targets under dynamic background. The experiment results show that the proposed targets separation method of the feature points integrated with the texture detection in this experiment can successfully separate dynamic targets from static targets based on ORB and SURF feature extraction methods under four different dynamic environment. It can also complete the optimization of the relationship between frame and frame. The proposed separation method of the feature points in the paper can obtain good effect in different dynamic environments.

The remainder of the paper is organized as follows. In section Related Work, the background studied in this paper and related work are introduced. The overview of the system is presented in section System Overview. The static feature detection and pose optimization algorithm proposed in the paper are detailed in section Static Feature Detection and Pose Optimization. The realization of dynamic target detection algorithm is introduced in section Constraint. In section Experiment Results, experiments and results in different environment are given. Concluding remarks are given in section Conclusion and Future Work.

RELATED WORK

Moving object detection is one of the hotspots in machine vision research (Zhang et al., 2014, 2016; Choi and Maurer, 2016; Naiel et al., 2017). Fischler and Bolles (1981) proposed a paradigm for model fitting with applications to image analysis and automated cartography. According to the status of a camera, it can be divided into the static background detection and the dynamic background detection. In the static background

detection, the camera is always stationary, so the moving target detection is easy. It has been widely used in scene monitoring of fixed environment, such as factory, road, and airport. The common background models have an adaptive background model based on the kernel density estimation, the Gaussian background model and the hidden Markov background model. In the dynamic background detection, the position of the camera changes, which can result in the change of the background and object in the image at the same time. So it increases the difficulty of moving target detection, which is the focus of the current research for moving target detection. There are three main categories about dynamic background detection of moving objects (Xu et al., 2011; Yin et al., 2016): optical flow, background comparison, and interframe difference method. For the optical flow method, because the background and the moving speed of the detected target is different, it can result in a large difference in optical flow. So moving objects can be identified according to it. The calculation of the optical flow is large, and there exists pore size problem. The comparison method of the background adopts image registration to dynamically update the background model, and it can compare the actual image with updated the background model to obtain the moving target. The interframe difference method is used to register the background of several successive images. The target detection is transformed into the moving object detection problem in the static background, and the moving object is separated by the difference image of the front and rear frames. The background image registration method includes texture algorithm, Fourier transform method, and feature matching method (Naiel et al., 2017). The feature matching method is simple and fast, but the matching error of the existing matching method is influenced by the changing environment.

Some previous research didn't consider the dynamic scene, and the detected dynamic or uncertain points were discarded as abnormal values (Williams et al., 2007; Paz et al., 2008; Liang and Min, 2013; Zhang et al., 2015; Zhou et al., 2015). However, when some of the dynamic objects are large, they would have large error, or even failure. In the dynamic environment, the existing research mainly adopts the filter approach (Hahnel et al., 2003) and has been successfully applied to solve the problems of SLAM based on laser scanner and radar system, but the research of applying to visual SLAM is seldom studied. Fang et al. (2019) detect and recognize the target through visual tactile fusion. Gao and Zhang (2017) explains the basics of SLAM. In Chen and Cai (2011), the visual sensor and the radar were used to detect the dynamic object. The uncertain factor is detected by using the eight-neighborhood rolling window method based on the map difference method. But the long-time static target is difficult to be detected out accurately. Zhang et al. (2018, 2019), Afiq et al. (2019) applied dynamic target detection to crowd action and emotion recognition. In addition, the lack of laser radar can bring instable judgment for some obstacles with special material (such as glass doors), which can affect the accuracy of the map (Einhorn and Gross, 2015), achieved estimation tracking of the dynamic object by combining the normal distribution transformation with the grid map. But there are some restrictions on the scope of adaptation. In Sun et al. (2017), a novel movement removal method based on RGB-D data was proposed, which enhanced

the application in dynamic environment. The work of Lee and Myung (2014) and Wang et al. (2016) used the posture structure diagram and the RGB-D sensor to complete the detection of the low speed target, and re-optimized the structure of posture to obtain the corrected location and mapping results. These dynamic separation methods are still not suitable very well for the detected target with speed and volume. And the judgment is still not accurate enough or they require a high economy. In Zou and Tan (2013), the cooperation of multiple cameras is used to detect dynamic points. The separation between static feature points and uncertain points is made by a camera at first. Then multiple cameras are used to further determine the uncertainty points. The method reduces the impact of the large-scale moving objects on the system, and it is suitable for high dynamic object. The method proposed in this paper is based on Zou and Tan (2013).

In this experiment, a camera is used to solve classification problem of the feature points. Comparing with other sensors, the camera is passive, compact, and energy-saving. It has an important advantage for intelligent platforms with limited weight and energy capacity. The proposed algorithm in this experiment can achieve the detection classification of the moving objects with different speeds in a variety of complex environment, and complete the pose transformation optimization only by a visual sensor.

SYSTEM OVERVIEW

In this experiment, a visual sensor can be used to obtain the environment information. The frame image obtained by the visual sensor is regarded as the input of the whole system framework. By calculating the transformation matrix between frame and frame, the dynamic feature points and static feature points are finally judged by the re-projection error. The framework of this system is divided into three parts shown in **Figure 1**. The first part is the initialized process, in which the feature of each frame image is extracted and the image relationship between frame and frame is calculated. The feature extraction of image frames has two methods. One is SURF feature extraction (Bay et al., 2006). SURF algorithm with robustness and fast speed is an enhanced version of the SIFT algorithm (Lowe, 2004). The other is ORB feature extraction algorithm (Rublee et al., 2011), which combines the FAST feature detection method with the BRIEF feature descriptor. It improves and optimizes the original two algorithms. It has a invariable scale, and the speed is faster 10 times than SURF. Different feature extraction is chosen according to the different scenes. After obtaining the feature points, the feature points can be matched by the existing Brute Force method. The matched result is passed to the homography matrix frame to obtain the transformation relation between frame and frame. By this transformation relation, the re-projection error is calculated to obtain an initialized static feature point and dynamic feature point. And the next two parts is based on the initial work.

In the paper, the dynamic and static region constraint methods are proposed for different feature attributes, and the texture detection is incorporated to realize screening data. The re-projection error method is used again when the dynamic feature point is judged, which is also the second re-projection error method proposed in this paper. It can realize the classification of dynamic target and the relationship optimization between frame and frame. The second part is filtered out static point in the second frame and the static region constraint is used to reduce false matching. The second judgment based on the texture detection is used to process the determined static point, which can achieve the separation of static points finally. After the separation of the static points, the exact static matching point is given to the homography matrix again, and then an optimal relation matrix between the frame and frame is obtained. Through this relation matrix, the uncertain points can be judged twice. The results will be used in the third part of the system. Thus, the separation of static points, the relationship optimization between frame and frame and the second judgment of uncertain points are finished, which realizes a rough judgment for the dynamic points. The third part is the separation of the dynamic feature points. The main modules are divided into dynamic region constraints and double texture detection fusion. Although the dynamic points and static points separation have the similar principle, the means of implementation are different. In the next section, the principles, methods, and parameters of each frame will be introduced.

STATIC FEATURE DETECTION AND POSE OPTIMIZATION

For each frame of the image, dynamic points and the static points from background image are distinguished. Because of the change of the three-dimensional position of the dynamic feature point, the projection point should be far away from the corresponding feature point when dynamic feature point of the *(n-1)*th frame is projected onto the *n*th frame picture. For the static feature point, the re-projection error should be small. Based on this principle, an algorithm is designed to distinguish dynamic feature points from static feature points.

Initialization

The first step of the proposed algorithm is the initialization process. The image feature of each frame is extracted by using the ORB and SURF feature extraction algorithm, respectively, under a variety of complex environment. For the both methods, the feature points extracted using ORB are less than that of SURF but the speed is faster. The two extraction methods can be selected according to application background and real-time requirements. The extracted feature points are used as the data source for the whole system. These data can be analyzed and the dynamic feature points and the static feature points can be distinguished.

Firstly, after extracting the image feature information of each frame, the data association of the adjacent frame is realized by using the existing matching algorithm, that is, the corresponding feature points between the two frames are found. For example, there is the feature point *A* in the *(n-1)*th frame, then the position of *A* will be at the position of the feature point *B* in the nth frame after one frame changes, which is shown in **Figure 2**. Although there exits the mismatching result, the matching result is not
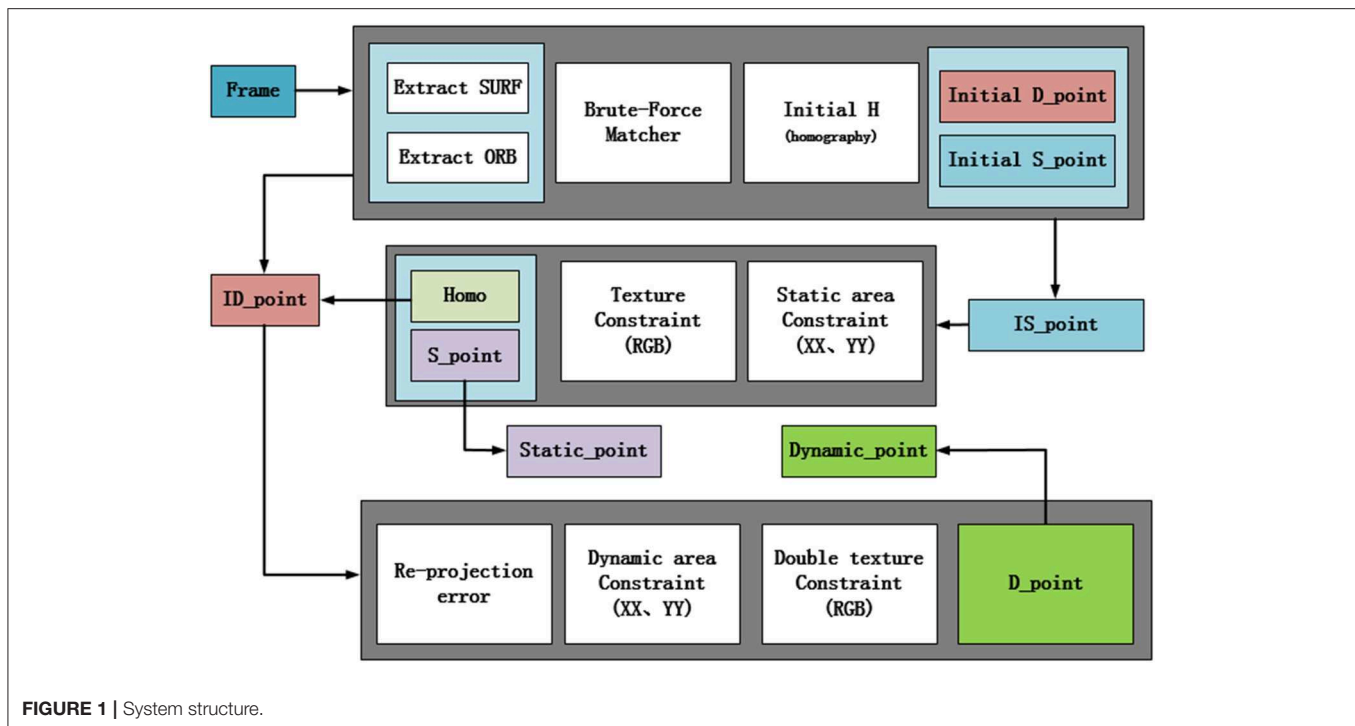
FIGURE 1 | System structure.
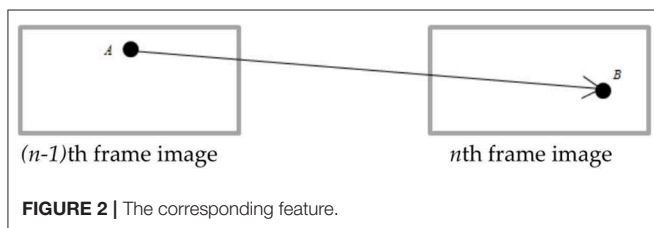
FIGURE 2 | The corresponding feature.
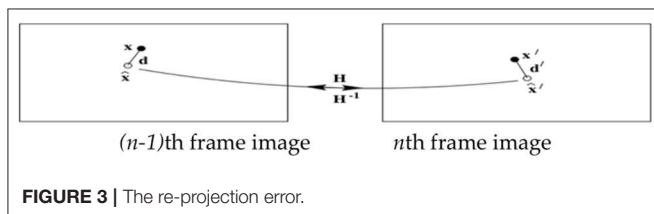
FIGURE 3 | The re-projection error.

processed here. The result is directly passed to the homography matrix. According to $P_2 = HP_1$, homography matrix between two frames is obtained easily by using RANSAC.

Then the static feature points and the dynamic feature points obtained by Brute force method will be separated by using the re-projection error method. The re-projection error is shown in the **Figure 3**. Its expression is as follows.

$$d(x,\hat{x})^2 + d(x',\hat{x}')^2 \qquad (1)$$

According to the re-projection error, the feature points extracted from the **(n-1)**th frame are projected onto the **n**th frame image, and the distance between the corresponding point and

the projection point of the **(n-1)**th frame image at the **n**th frame image is compared. The feature point obey the Gaussian distribution. The Mahalanobis distance between the projection point and the corresponding point is less than a certain value. If the feature point is greater than the Mahalanobis distance, this feature point may be a dynamic point or a mismatch point, which is called an uncertain point. If the feature point is less than the Mahalanobis distance, this feature point may be a static point. It will be regarded as an initial value of a static point which is prepared for the next step. So the whole initialization process is completed, and the process is shown in **Table 1**.

Texture Constraint

In adjacent frames, the static point and its re-projection point change very little, and the texture difference is very small, while the dynamic point and its re-projection point change greatly, and the texture difference is large.

The interframe difference method is used to register the background of several successive images, compensate the difference of moving background. The problem of target detection is transformed into the problem of moving target detection in static background, and the moving object is separated by the difference images of adjacent frames. Let the binary image $I_{k-1}(x,y)$ and $I_k(x,y)$ be the registered image of the two adjacent frames, and the binary image of the moving object can be obtained by processing the interframe difference and threshold.

$$d_{k-1,k}(x,y) = \begin{cases} 1, & |I_k(x,y) - I_{k-1}(x,y)| > T \\ 0, & otherwise \end{cases} \qquad (2)$$

Initialization process (dynamic feature points, static feature points, the relationship between frame and frame)
Feature point extraction: keypoints_1,keypoint_2
Feature descriptor: descriptors_1,descriptors_2
Feature matching by Brute Force method: good_matches
The Homography matrix is obtained by giving the matching result to the Homography matrix frame: Initial_***H***
Initial_H(that is, the relationship between the frame and the frame is initialized)
To find the projection point by ***H***
Compare the projection point with the corresponding point of Maha distance: diff
Define an error threshold: ***θ***
For *i*=1: *diff.rows*
If (*diff* < *θ*)
The feature point is a static point
—>initial_S_point
Else
The feature point is the dynamic point
—>Initial_D_point
End for

Theoretically in the difference binary image $d_{k-1,k}(x,y)$, only the pixel position in the area covered by the moving object is not zero. However, in practice, because there exist high frequency noise, illumination change and the subtle change of the background in the image and registration will also cause errors at the same time, many pixels are non-zero in the binary image besides moving objects, that is, there exist noisy points. And the method can only segment the contours of the moving objects.

In this paper, frame difference method is used to complete the texture constraint of feature points. The collected picture is RGB three-channel map, the texture constraint will be completed by using three-channel (RGB). The two adjacent frames are sampled in three channels, and the different weighted processing is used according to the sensitivity of each channel. The weighting ratio is set to be 1 and sum operation is done. Let the threshold value be *T*. If the value of three-channel doing the difference and weighted summing is greater than this threshold *T*, it demonstrates that the two points do not correspond to the same point. Otherwise,

the two points correspond to the same point. The equation is as follows.

$$d_n(x,y) = \alpha(|R_n(x,y) - R_{n-1}(x,y)|) + \beta(|G_n(x,y) - G_{n-1}(x,y)|) + \gamma(|B_n(x,y) - B_{n-1}(x,y)|) \quad (3)$$

$$\alpha + \beta + \gamma = 1 \quad (4)$$

If $d < T$, it means that the two points correspond to the same spatial point, and vice versa, it means that the two points correspond not to the same spatial point. Where d_n is the texture error of two frames, α, β, γ are the weight parameters.

The original frame difference method is no longer used to traverse the whole image pixel, but the feature point.
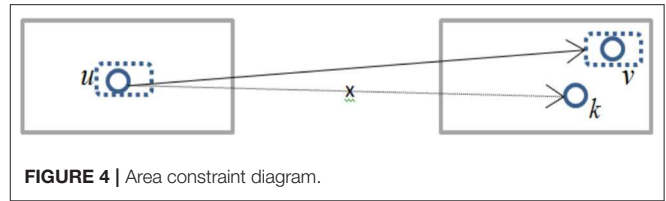
FIGURE 4 | Area constraint diagram.

Compared with million pixels of each frame, the feature points are only a few thousand or hundred of feature points, the amount of the calculation is greatly reduced. So the designed texture constraints can be integrated into the feature point classification system proposed in this paper.

Area Constraint

In order to reduce the mismatch, a region constraint method is designed based on the principle of the constraint line in this paper (Zou and Tan, 2013). Block constraint is mainly used. Assume that there are two candidate points on the *nth* frame image corresponding to the *(n-1)th* frame image. In this paper, a constraint window is set, which is shown in **Figure 4**.

When *v* and *u* are in the same window, *v* and *u* are matched each other. When *k* is beyond this window, *k* and *u* do not match. In order to adapt a variety of environment, the size of this active window is changed. It not only improves the accuracy of matching, but also ensures the matching points not to be too sparse. The difference square of the pixel coordinates *u* and *v* between each matching point of the adjacent image frame are solved when a window is created. Because the difference between the matching point may be negative, the square operation is added. The maximum square of difference value between the adjacent frame *u* and *v* is obtained by traversing the image frame, respectively. The active window with region constraint is built by obtaining the maximum value, and the length and width of the window are determined by the maximum value of *u* and *v,* respectively. The expression is as follows:

$$f(P_n, P_{n-1}) = \begin{cases} \text{Correct correspondence} & |u_n - u_{n-1}| < \theta U_{\max} \text{ and } v_n - v_{n-1}| < \eta V_{\max} \\ \text{Error correspondence} & otherwise \end{cases} \quad (5)$$

Where $P_n(u_n,v_n)$ and $P_{n-1}(u_{n-1},v_{n-1})$ are the corresponding points between the adjacent frames, U_{max} and V_{max} are the square of the maximum difference, θ and η are the proportional coefficient. In Equation (5), a window with length U_{max} and width V_{max} is established which realizes the region constraint for the matching point. Here, the size of the active window is automatically adjusted according to corresponding points between frame and frame influenced by the noise, while the proportional coefficient is set in order to ensure the sparseness between frame and frame. So the whole system automatically adjusts the size of the window in different environment, which can ensure the adaptability of the system.

The separation process of static feature points

Initialize static feature points: Initial_S_point
Static Constraints:
Set up θ, η
Traverse the image U_{max}, V_{max}
  For $i=1$: *matches.size()*
    If the difference square of the adjacent coordinate
    points
      $XX^2 < \theta U_{max}, YY^2 < \eta V_{max}$
      Two points correspond
      Generate a new match pair: matches1
    Else
      If two points are mismatched, eliminated
    End if
  End for
Texture Constraints:
Set up α, β, γ
Traverse the image to get the texture error d_n
    For $i=1$: *matches.size()*
      If $d_n < T$
        Two points correspond
        Generate a new match pair matches2, get the
        final static feature point
    Static_point
      Else
        If two points are mismatched, eliminated
      End if
    End for
The final static match point is given to the homography matrix
Perform RANSAC algorithm
Get the conversion relationship between frame and frame Homography

Static Feature Detection

According to the initialization process in section Initialization, the initial homography matrix **H**, the initial static points and the uncertain points can be obtained. In this section, the initial static feature points are filtered by using the proposed method. The spatial position represented by the static point does not change, so the interference is very small in the short frame time. The change of three color channels is also very small, so the judgment is more accurate. The overall process of the static target is shown in **Table 2**.

First, the initialized static point is used as input. Because the attributes of static features do not change much, the static region constraints are combined with the region constraints module in section Initialization. And the region constraint window is relatively small, which is more accurate for mismatches. However, there are still some mismatches, and the screening results can not meet the actual requirements. Therefore, this paper introduces the method of texture constraints.

The three color channels can greatly reduce the error caused by illumination because of the change of pose. Secondly, the whole frame is processed by a frame and a frame. The pose of the static target is changed a little in the view angle. So the texture constraint is added, which can greatly reduce the mismatching result. According to the two screening steps proposed in this paper, a set of accurate static feature points can be obtained. Then the extraction of static feature points is completed. The extracted static points can be pass to the homography matrix

FIGURE 5 | Double texture.

again and the homography matrix is solved by the RANSAC algorithm. In this case, the homographic matrix **H** can accurately express the transformational relation between the two frames, and the homographic matrix can be decomposed to obtain the transformation matrix **R** and translation matrix **T** between the frame and frame. It can achieve the function of visual odometer. But in this experiment, only the change of the relationship between the two frames is needed, so the homographic matrix **H** doesn't need to be decomposed. The homography matrix obtained again will be used in the dynamic target separation.

CONSTRAINT

Double Texture Constraint

The texture problem of dynamic feature points is different from that of static feature detection. Each dynamic point has two points, one is the projection point, the other is the matching point. In **Figure 5**, i is a dynamic feature point, then j is the point which i is projected by the homography matrix in section Static Feature Detection, and p is obtained by descriptor matching of the adjacent two frames.

In the two frame images, i and j correspond to the same spatial position. The dynamic point i is moved from the projection position j to p. The difference between i point of the **(n-1)**th frame and the **p** point of the **n**th frame is small, while the difference of the texture between the i and j points is very big (although sometimes the background is similar to the texture of the moving object). According to this principle, a double texture constraint is designed to separate dynamic feature points by the texture constraint in section Area Constraint.

According to section Texture Constraint, the first texture constraint is the same as the static texture constraint, and the texture check between matching points is done. In the second texture constraint, the texture constraint between the re-projection points is compared. The expression is as follows.

$$dd_n = \alpha(|i(x,y) - j(x,y)|) + \beta(|i(x,y) - j(x,y)|) + \gamma(|i(x,y) - j(x,y)|) \quad (6)$$

$$f(x,y) = \begin{cases} \text{Dynamic point} & d_n < T \text{ and } dd_n > T \\ \text{Error correspondence} & otherwise \end{cases} \quad (7)$$

Where dd_n is the dynamic texture error of two frames, $f(x,y)$ completes the judgment of the nature of the point (x,y).

In **Figure 7**, $j(x,y)$ is the projection point of the previous frame image $i(x,y)$, where the coordinates j can be calculated from the obtained homography matrix. Other parameters can be given by static texture constraint.

TABLE 3 | The separation process of the dynamic target.

The initialization process of the feature points
Initial value: ID_point Homography
Second re-projection error
Double texture constraint
Set up α, β, γ
Traverse the image to get the texture error d_n, dd_n
For $i=1$: *matches.size()*
If $d_n < T$ and $dd_n > T$
Get the final static feature point Dynamic_point
Else
If two points are mismatched, eliminated
End if
End for

FIGURE 6 | Bumblebee binocular camera.

Dynamic Detection

This section will introduce the separation of the dynamic feature points. The initialization of the dynamic feature points can be obtained by the initialization process in section Initialization, and the homographic matrix H between frame and frame can be obtained by separating the static feature points. The two data sets are regarded as the input values of the dynamic separation. The initial dynamic points contain the dynamic points and the mismatch points. In this paper, the mismatch points will be filtered by the proposed method in section Double Texture Constraint and the accurate dynamic points are separated. Firstly, uncertain points can be filtered out by the second re-projection error method proposed in this paper. The accurate dynamic feature points can be obtained because the filtering is performed by an accurate homographic H. But the accuracy is not good enough to cause the converge. So a dynamic region constraint method similar to the static region constraint is introduced. But the dynamic feature points in the actual space will change greatly, so the ratio parameters of regional window will be changed greatly. Otherwise the real dynamic feature points will be regarded as mismatch points and removed, and the separation of dynamic points can not be completed. The size of the window is decided by the distance of all dynamic feature points, which can ensure a certain number of dynamic points. Some of the dynamic points that does not meet the region are removed by the constraints of dynamic region, but some of the error points are still retained. According to the dual texture constraint method in section Double Texture Constraint, the three color channels are performed weighted constraint at the same time. Uncertain points and their own projection points are compared with the texture of the corresponding point. This method can realize the interaction of features and texture, and achieve the separation of dynamic feature points. The process is shown in **Table 3**.

Both the separation of the static points in section Static Feature Detection and Pose Optimization and the separation of the dynamic points in section Experiment Results use the region constraint and texture constraint. Although the two separation is similar in principle, the implementation method is different. The two constraint methods in the static region play an important role, while the filtering effect of the region constraint in the dynamic separation is not obvious, which is mainly caused by the attribution of the dynamic points. Because the feature matching coordinate is computed by the two constraint methods and the descriptor is not involved, so the computation time is not long. The experiment will be performed in the next section.

EXPERIMENT RESULTS

The computer operated in the experiment is Windows 64-bit, operating system AMD A6-6400 APU with Radeon (tm) HD Graphics (3.90 GHz), 4GB RAM. The visual sensor in this paper is used Bumblebee binocular camera shown in **Figure 6**. Please note that Bumblebee has two cameras, but only one of them is used in the experiment.

The frame sequence of the environment is obtained off-line. The resolution of the frame image is $1,024 \times 768$, 15 frames/s. The obtained image frame is directly used in the experiment. Four experiment scene is designed to test the effect of the extracted feature by using the proposed method based on ORB and SURF. The blue dot in the experiment represents the static points obtained by the experiment and the red dot represents the dynamic points, which are shown in the video.

Experiment 1: An object with a large area moves in a static background. In the experiment, the camera keeps stationary, that is, the environment background does not change, and the person moves as a dynamic target at a relatively close distance of the camera, the close distance is within 1 m. In this scene, the classification effect of the environment feature is tested by the proposed method in this paper when the object with a large area (the object with a large area is the person in this experiment) moves close to the camera.

In **Figure 9**, the image on the left is the experiment result of the classification by using the proposed method based on ORB, and the image on the right is the experiment result of the classification by using the proposed method based on SURF. At the same time, the average consuming time of each step is summarized, which is shown in **Table 4**. The extraction of the environment feature is completed by using the second re-projection error integrated with the texture detection based on ORB algorithm and SURF algorithm. The experiment results show that the two proposed methods in this paper can complete the Feature classification of the environment feature. **Table 4** shows the speed of the feature classification by using the second

TABLE 4 | The experiment 1 results using the proposed method based on ORB and SURF.

Time (ms/frame)	Feature point extraction	Feature point matching	Feature classification
ORB	527.17	26.57	10.96
SURF	8378.05	422.55	24.88

FIGURE 7 | An object with a large area moves close to the camera in the static background. **(A)** The proposed method based on ORB. **(B)** The proposed method based on SURF.

FIGURE 8 | Multiple moving objects with far distance in a static background (1). **(A)** ORB method. **(B)** SURF method (One student). **(C)** ORB method. **(D)** SURF method (One student).

FIGURE 9 | Multiple moving objects with far distance in a static background (2). **(A)** ORB method. **(B)** SURF method (Three student). **(C)** ORB method. **(D)** SURF method (Four student).

TABLE 5 | The experiment 2 results using the proposed method based on ORB and SURF.

Time (ms/frame)	Feature point extraction	Feature point matching	Feature classification
ORB	794.78	26.46	9.86
SURF	8229.56	362.00	23.24

re-projection error integrated with the texture detection method based on ORB is much faster than that of the proposed method based on SURF. However, **Figure 7** shows the classification error by using the proposed method based on ORB is smaller than that of the proposed method based on SURF.

Experiment 2: Multiple objects in the far distance move in a static background, the far distance is about 5–8 m. The longest distance in the experiment is about 8 meters, which can ensure the experiment effect. In the experiment, the camera remained stationary, so the environment background also remains stationary. Four students in the far distance walk through the camera in turn, while the camera collects the

dynamic scene with far distance from one dynamic object to multiple dynamic objects. When the crowd has displacement in the plane direction of the camera, the effect of this method is acceptable, but when the direction of crowd movement is perpendicular to the camera, the moving person may be misjudged as a static point.

Figures 8A,C, **9A,C** are the experiment results in the far distance from one dynamic object to multiple dynamic objects by using the proposed method based on ORB. **Figures 8B,D**, **9B,D** are the experiment results in the far distance from one dynamic object to multiple dynamic objects by using the proposed method based on SURF. At the same time, the average consuming time of each step is summarized, which is shown in **Table 5**. The experiment results demonstrate that the proposed method in this paper can realize the feature classification whether or not there exist one or multiple moving objects with far distance in a static background. And **Table 5** shows the speed of each step for Experiment 2.

Experiment 3: An object with a large area moves in a dynamic background. The pose of the camera changes when the camera extracts the frame image. The camera and the moving object can move without the limitation of position and speed.

The experimental results are shown in **Figure 9**. **Figures 10A,C** are the experiment results of the classification by using the proposed method based on ORB, and **Figures 10B,D**

FIGURE 10 | One object moves close to the camera in the dynamic background. **(A)** ORB method. **(B)** SURF method. **(C)** ORB method. **(D)** SURF method.

FIGURE 11 | Multiple moving objects with far distance in a dynamic background. **(A)** ORB method. **(B)** SURF method.

TABLE 7 | The experiment 4 results using the proposed method based on ORB and SURF.

Time (ms/frame)	Feature point extraction	Feature point matching	Feature classification
ORB	645.76	24.60	91.87
SURF	10052.98	512.76	131.22

TABLE 6 | The experiment 3 results using the proposed method based on ORB and SURF.

Time (ms/frame)	Feature point extraction	Feature point matching	Feature classification
ORB	627.58	24.47	84.03
SURF	7767.62	265.42	106.13

are the experiment result of the classification by using the proposed method based on SURF. The **Figures 10A,B** are the results of the initial state of the environment by using the proposed method based on ORB and SURF in this paper. At the same time, the average consuming time of each step is summarized, which is shown in **Table 6**. The experimental results demonstrate that both of the methods can identify the static point. **Figures 10C,D** is the experiment results of one dynamic object by using the proposed method based on ORB and SURF in the paper. Both of the two proposed methods based on ORB and SURF can successfully identify the static and dynamic feature points. **Figure 10** shows the classification feature by using the proposed method based on SURF is more than that of the proposed method based on ORB. And **Table 6** shows the speed of the feature classification using the two proposed methods based on ORB and SURF.

Experiment 4: Multiple objects in the far distance move in a dynamic background. The pose of the camera changes when the camera extracts the frame image. The camera and all the moving objects can also move without the limitation of position and speed.

The experimental results are shown in **Figure 11**. **Figure 11A** is the experiment result of the classification by using the proposed method based on ORB, and **Figure 11B** is the experiment result of the classification by using the proposed
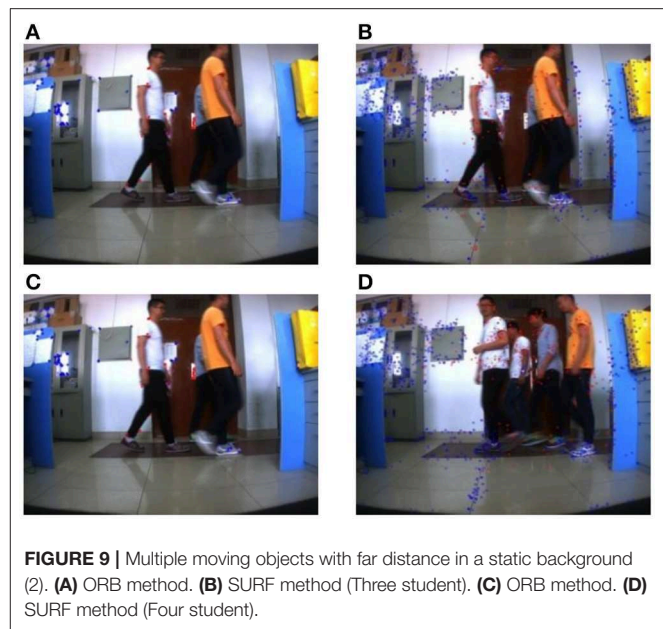
method based on SURF. At the same time, the average consuming time of each step is summarized, which is shown in **Table 7**. **Figure 11** shows the experiment results show that both of the two proposed methods can complete the feature classification of the environment. However, **Table 7** shows the speed of the classification by using the second re-projection error integrated with the texture detection method based on ORB is much faster than that of the proposed method based on SURF.

The experiments show that the feature points can be classified accurately for one object or multiple objects whether or not in the static environment or in the dynamic environment.

In addition, the number of static points and dynamic points are counted based on the classification of each frame image in experiment 2 (shown in **Figure 12**) and experiment 4 (shown in **Figure 13**). The consumed time for each frame classification algorithm (excluding feature extraction and matching time) is also compared. Here the time of the feature points processed is considered because the classification algorithm is proposed in this paper. Feature extraction and matching are the application of existing methods. At the same time, the average consuming time of each step in experiment 4 is summarized, which is shown from **Table 4** to **Table 7**. Then the real-time performance of the four environment by using the proposed method in the paper can be compared.

In Zou and Tan (2013), the KLT tracker is used to complete the feature extraction by using multiple visual sensors. And the feature classification is realized by using the re-projection error under the GPU acceleration, which is more suitable for the feature detection of the high-speed dynamic environment. The results in Zou and Tan (2013) is shown in **Figure 14**. This paper is based on the study of Zou and Tan (2013). Different from Zou and Tan (2013), this paper only uses a sensor to achieve the feature classification, which is more practical and
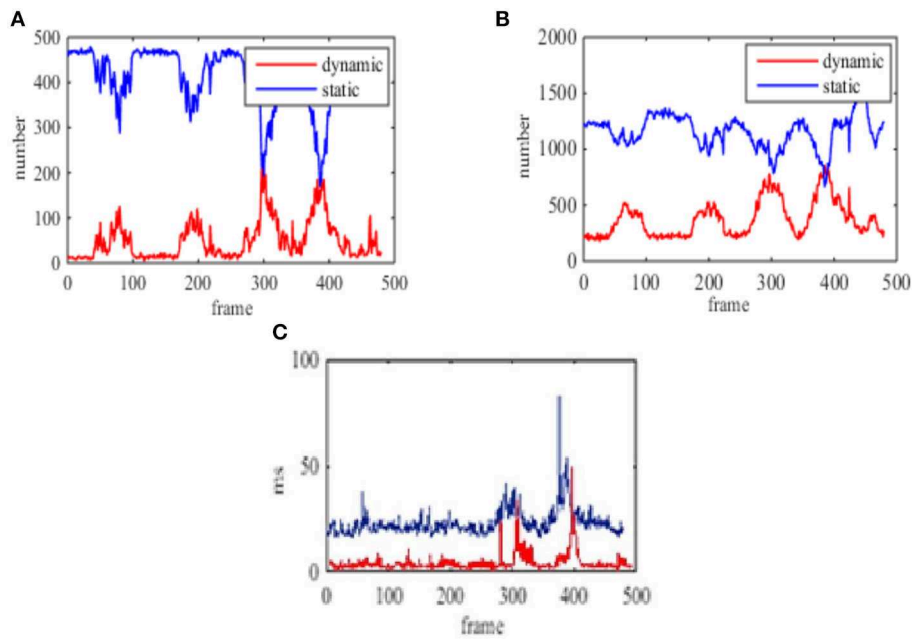
FIGURE 12 | The number of feature points of multiple moving objects in the far distance and the time of each frame in the static background. **(A)** The classification number of the static and dynamic points based on ORB. **(B)** The classification number of the static and dynamic points based on SURF. **(C)** The processing time based on ORB and SURF, Red curve is ORB, blue curve is SURF.
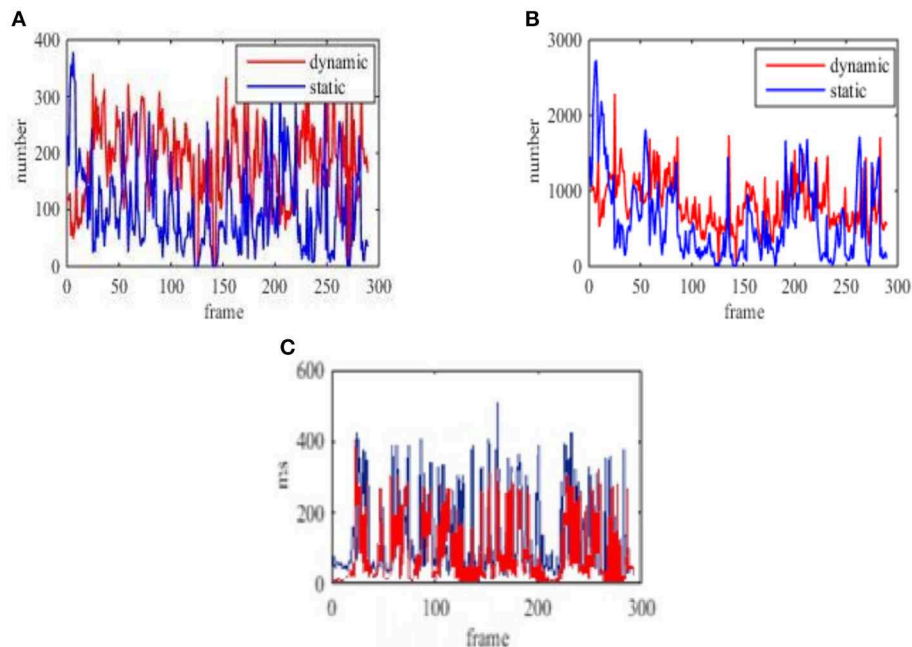
FIGURE 13 | The number of feature points of multiple moving objects in the far distance and the time of each frame in the dynamic background. **(A)** The classification number of the static and dynamic points based on ORB. **(B)** The classification number of the static and dynamic points based on SURF. **(C)** The processing time based on ORB and SURF, red curve is ORB, blue curve is SURF.

convenient for the SLAM. **Figures 12**, **14A** shows the number of classification is much more than that of Zou and Tan (2013) when SURF is used to extract the feature of the environment.

Although, the number of feature classification using ORB to extract the feature of the environment is less than that of using SURF, the running time using ORB is only a dozen

FIGURE 14 | The experiment results of Zou and Tan (2013). **(A)** The classification number of the static and dynamic points in Zou and Tan (2013). **(B)** The running time of the static and dynamic points in Zou and Tan (2013).
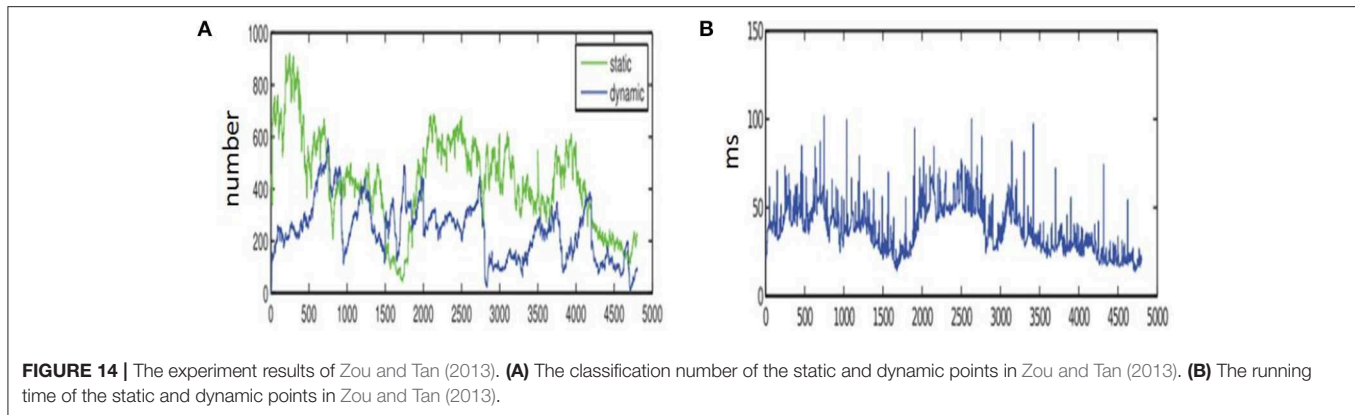
milliseconds. **Tables 4–7** shows that the time of processing feature extraction using SURF is 130 ms average per-frame. But the time of processing feature extraction using ORB is 91.87 ms average per-frame. At the same time, **Figures 12**, **14B** also shows it conforms to the require of real-time just as Zou and Tan (2013). The proposed algorithm in this paper can adapt to the dynamic environment with all kinds of speed, and has good effect for the moving objects with large area. So the second re-projection error method integrated with the texture detection proposed in the paper can obtain good performance for the feature classification of the dynamic environment by using a vision sensor.

Figures 12A,C show the number of feature points and processing time for each frame using the proposed method based on ORB in the static background. **Figures 12B,C** show the number of feature points and processing time for each frame using the proposed method based on SURF in the static background.

Figures 13A,C show the number of feature points and processing time for each frame using the proposed method based on ORB in the dynamic background. **Figures 13B,C** show the number of feature points and processing time for each frame using the proposed method based on SURF in the dynamic background.

CONCLUSION AND FUTURE WORK

This paper presented a new classification algorithm of feature points in dynamic environment by using a visual sensor. The designed three framework is initialization, static targets detection and dynamic targets detection, which can complete the classification of dynamic feature points and static feature points in various complex environments. And the transformation relationship between frame and frame is optimized. The experiment results demonstrate that the feature points can be classified accurately for one object or multiple objects whether or not in the static environment or in the dynamic environment. And the proposed algorithm in this paper has good effect for the moving objects with large area. So the

second re-projection error method integrated with the texture detection proposed in the paper can obtain good performance for the feature classification of the dynamic environment by using a vision sensor. The experiment was carried out indoors, and the method can work normally under sufficient sunshine or light. In order to improve the illumination adaptability of this method, RGB three-channel can be optimized to RGB-HSV six-channel in subsequent experiments, and the weight of H-channel can be increased to improve the illumination adaptability.

In future work, we will further study the image processing and screening, and apply the proposed algorithm to the SLAM.

DATA AVAILABILITY STATEMENT

The datasets for this article are not publicly available because: [The dataset contains some private information]. Requests to access the datasets should be directed to SWe, swen@ysu.edu.cn.

ETHICS STATEMENT

Written informed consent was obtained from the individual(s) for the publication of any potentially identifiable images or data included in this article.

AUTHOR CONTRIBUTIONS

SWe and ZZ put forward this idea and made a rigorous theoretical analysis. Under the guidance of SWe, ZZ completed the experiment and collected data. SWa, XZ, and DZ compiled and analyzed the experimental results and data to form the final manuscript. All authors provided critical feedback and made improvements and optimizations.

FUNDING

REFERENCES

Afiq, A. A., Zakariya, M. A., Saad, M. N., Nurfarzana, A. A., Khir, M. H. M., Fadzil, A. F., et al. (2019). A review on classifying abnormal behavior in crowd scene. *J. Vis. Commun. Image Represent.* 58, 285–303. doi: 10.1016/j.jvcir.2018.11.035

Bay, H., Tuytelaars, T., and Gool, L. V. (2006). SURF: speeded up robust features. *Comp. Vision Image Understand.* 110, 404–417. doi: 10.1007/11744023_32

Chen, B. F., Cai, Z. X, and Zou, Z. R. (2011). Design and implementation of mobile robot SLAMiDE system. *Comp. Eng.* 37, 158–160. doi: 10.3969/j.issn.1000-3428.2011.17.053

Choi, J., and Maurer, M. (2016). Local volumetric hybrid-map-based simultaneous localization and mapping with moving object tracking. *IEEE Trans. Intell. Transport. Syst.* 17, 2440–2455. doi: 10.1109/TITS.2016.2519536

Einhorn, E., and Gross, H. M. (2015). Generic NDT mapping in dynamic environments and its application for lifelong SLAM. *Robot. Auton. Syst.* 69, 28–39. doi: 10.1016/j.robot.2014.08.008

Fang, B., Sun, F., Liu, H., Tan, C., and Guo, D. (2019). A glove-based system for object recognition via visual-tactile fusion. *Sci. China Inf. Sci.* 62:050203. doi: 10.1007/s11432-018-9606-6

Fischler, M. A., and Bolles, R. C. (1981, June). Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM* 24, 381–395.

Gao, X., and Zhang, T. (2017). *VSLAM: From Theory to Practice.* Beijing: Electronic Industry *Press*, 146–148.

Hahnel, D., Triebel, R., Burgard, W., and Thrun, S. (2003). Map building with mobile robots in dynamic environments," in *IEEE International Conference on Robotics and Automation, 2003. Proceedings. ICRA'03* (Taipei), 1557–1563.

Klein, G., and Murray, D. (2007). "Parallel tracking and mapping for small AR workspaces," in *IEEE and ACM International Symposium on Mixed and Augmented Reality* (Nara: IEEE Computer Society), 1–10.

Lee, D., and Myung, H. (2014). Solution to the SLAM problem in low dynamic environments using a pose graph and an RGB-D sensor. *Sensors* 14, 12467–12496. doi: 10.3390/s140712467

Liang, M. J., Min, H. Q, and Luo, R. H. (2013). A survey of simultaneous location and map creation based on graph optimization. *Robot* 35, 500–512. doi: 10.3724/SP.J.1218.2013.00500

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *Int. J. Comp. Vision* 60, 91–110. doi: 10.1023/B:VISI.0000029664.99615.94

Mur-Artal, R., Montiel, J. M. M., and Tardos, J. D. (2015). ORB-SLAM: a versatile and accurate monocular SLAM system. *IEEE Trans. Robot.* 31, 1147–1163. doi: 10.1109/TRO.2015.2463671

Mur-Artal, R., and Tardos, J. D. (2016). ORB-SLAM2: an open-source SLAM system for monocular, stereo, and RGB-D cameras. *IEEE Trans. Robot.* 33, 1255–1262. doi: 10.1109/tro.2017.2705103

Naiel, M. A., Ahmad, M. O, Swamy, M. N. S, Lim, J., and Yang, M.-H. (2017). Online multi-object tracking via robust collaborative model and sample selection. *Comp. Vision Image Understand.* 154, 94–107. doi: 10.1016/j.cviu.2016.07.003

Paz, L. M., Piniés, P., Tardós, J. D., and Neira, J. (2008). Large-scale 6-DOFSLAM with stereo–in -hand. *IEEE Trans. Robot.* 24, 946–957. doi: 10.1109/TRO.2008.2004637

Rublee, E., Rabaud, V., Konolige, K., and Bradski, G. (2011). "ORB: an efficient alternative to SIFT or SURF," in *International Conference on Computer Vision* (Barcelona: IEEE Computer Society), 2564–2571.

Saeedi, S., Trentini, M., Seto, M., and Li, H. (2016). Multiple-robot simultaneous localization and mapping: a review. *J. Field Robot.* 33, 3–46. doi: 10.1002/rob.21620

Sun, Y., Liu, M., and Meng, M. Q. H. (2017). Improving RGB-D SLAM in dynamic environments: a motion removal approach. *Robot. Auton. Syst.* 89, 110–122. doi: 10.1016/j.robot.2016.11.012

Wang, Y., Huang, S., Xiong, R., and Wu, J. (2016). A framework for multi-session RGBD SLAM in low dynamic workspace environment. *Caai Trans. Intell. Technol.* 1, 90–103. doi: 10.1016/j.trit.2016.03.009

Wen, S., Chen, X., Ma, C., Lamab, H. K., and Hua, S. (2015). The Q-learning obstacle avoidance algorithm based on EKF-SLAM for NAO autonomous walking under unknown environments. *Robot. Auton. Syst.* 72, 29–36. doi: 10.1016/j.robot.2015.04.003

Williams, B., Klein, G., and Reid, I. (2007). "Real-time SLAM relocalisation," in *International Conference on Computer Vision* (Oxford: IEEE), 1–8. doi: 10.1109/iccv.2007.4409115

Xu, T., Wang, C., Wang, Y., and Zhang, Z. (2011). "Saliency model based head pose estimation by sparse optical flow," in *The First Asian Conference on Pattern Recognition* (Beijing), 575–579. doi: 10.1109/acpr.2011.6166668

Yin, H. P., Chen, B., and Chai, Y. (2016). A survey of visual - based target detection and tracking. *Zidonghua Xuebao/acta Automatica Sinica* 42, 1466–1489. doi: 10.16383/j.aas.2016.c150823

Zhang, G., Jin, H. L, Lim, J., and Suh, I. I. H. (2015). Building a 3-D line-based map using stereo SLAM. *IEEE Trans. Robot.* 31, 1364–1377. doi: 10.1109/TRO.2015.2489498

Zhang, G. M., Zhang, S., and Chu, J. (2014). A new object detection algorithm using local contour features. *Zidonghua Xuebao* 40, 2346–2355. doi: 10.3724/SP.J.1004.2014.02346

Zhang, X., Shu, X., and He, Z. (2019). Crowd panic state detection using entropy of the distribution of enthalpy. *Phys. A Stat. Mech. Appl.* 525, 935–945. doi: 10.1016/j.physa.2019.04.033

Zhang, X., Sun, X., Xu, C., and Baciu, G. (2016). Multiple feature distinctions based saliency flow model. *Pattern Recogn.* 54, 190–205. doi: 10.1016/j.patcog.2015.12.014

Zhang, X., Yu, Q., and Yu, H. (2018). Physics inspired methods for crowd video surveillance and analysis: a survey. *IEEE Access* 6, 66816–66830. doi: 10.1109/access.2018.2878733

Zhou, H., Zou, D., Pei, L., Ying, R., Liu, P., and Yu, W. (2015). StructSLAM: visual SLAM with building structure lines. *IEEE Trans. Vehicul. Technol.* 64, 1364–1375. doi: 10.1109/TVT.2015.2388780

Zou, D., and Tan, P. (2013). CoSLAM: collaborative visual SLAM in dynamic environments. *IEEE Trans. Pattern Anal. Mach. Intell.* 35, 354–366. doi: 10.1109/TPAMI.2012.104

An Intuitive End-to-End Human-UAV Interaction System for Field Exploration

Ran Jiao[1], Zhaowei Wang[1], Ruihang Chu[1], Mingjie Dong[2], Yongfeng Rong[1] and Wusheng Chou[1,3*]

[1] School of Mechanical Engineering and Automation, Beihang University, Beijing, China, [2] College of Mechanical Engineering and Applied Electronics Technology, Beijing University of Technology, Beijing, China, [3] The State Key Laboratory of Virtual Reality, Technology and Systems, Beihang University, Beijing, China

This paper presents an intuitive end-to-end interaction system between a human and a hexacopter Unmanned Aerial Vehicle (UAV) for field exploration in which the UAV can be commanded by natural human poses. Moreover, LEDs installed on the UAV are used to communicate the state and intents of the UAV to the human as feedback throughout the interaction. A real time multi-human pose estimation system is built that can perform with low latency while maintaining competitive performance. The UAV is equipped with a robotic arm, kinematic and dynamic attitude models for which are provided by introducing the center of gravity (COG) of the vehicle. In addition, a super-twisting extended state observer (STESO)-based back-stepping controller (BSC) is constructed to estimate and attenuate complex disturbances in the attitude control system of the UAV, such as wind gusts, model uncertainties, etc. A stability analysis for the entire control system is also presented based on the Lyapunov stability theory. The pose estimation system is integrated with the proposed intelligent control architecture to command the UAV to execute an exploration task stably. Additionally, all the components of this interaction system are described. Several simulations and experiments have been conducted to demonstrate the effectiveness of the whole system and its individual components.

Keywords: UAV, intuitive interaction, pose estimation, super-twisting, extended state observer, back-stepping

1. INTRODUCTION

UAVs, which have been increasingly used as human assistants in various contexts in recent years, are developing very rapidly. They can be applied in areas to which humans cannot reach, such as for aerial photography, field exploration, etc. Also, human-robot interaction (Fang et al., 2019) has also been focused on recently, including human-UAV interaction technology. However, a traditional approach to the interaction between UAVs equipped with remote devices and a human is not convenient when that human is busy with other tasks during field exploration. This paper aims to build an intuitive end-to-end human-UAV interaction system for field exploration where mutual attention between the human and UAV is established in the process.

The interface used to control UAVs is an important part of the whole interaction system. It can be classified into two kinds, traditional human-computer interfaces and direct interfaces. As to the former, Rodriguez et al. (2013) designed ground control station software that is fully based on

open-source libraries and developed it for a platform composed of multiple UAVs for surveillance missions. Moreover, utility software designed by McLurkin et al. (2006) for interacting with hundreds of autonomous robots without having to handle them individually enables centralized development and debugging. In addition, several principles of swarm control are studied in Kolling et al. (2012) and are used in a simulated robot environment to enable a human operator to impose on and control large swarms of robots. Of the direct interfaces, many of them have been applied in human-UAV interaction systems in recent years. Pourmehr et al. (2013b) presents a multi-model system to create, modify, and command groups of robots, in which groups of robots can be created by speaking their numbers. Additionally, a whole system in which multiple humans and robots could interact with each other using a combination of sensing and signaling modalities was built by Pourmehr et al. (2013a). In our work, we use the direct interaction mode for the design of a natural and intuitive human-UAV interaction system as an assistant for field exploration. Similar to the interaction system mentioned by Monajjemi et al. (2013), human poses are used to give commands to the UAV in our interaction system. Therefore, the human detection system should be built first.

We intend to use several different natural human poses to communicate with the UAV. Previous research has looked into detecting serial human poses. A method based on Lagrangian particle trajectories, which are a suite of dense trajectories obtained by advecting optical flow over time, is proposed to capture the ensemble motions of a scene by Wu et al. (2011). Moreover, Bin et al. (2018) proposes a novel data glove for pose capturing and recognition based on inertial and magnetic measurement units (IMMUs). Additionally, Ran et al. (2007) proposes two related strategies. The first estimates a periodic motion frequency with two cascading hypothesis testing steps to filter out non-cyclic pixels, and the second involves converting the cyclic pattern into a binary sequence by fitting the Maximal Principal Gait Angle. Pishchulin et al. (2016) proposes a method to jointly solve the tasks of detection and pose estimation in which the number of persons in a scene can be inferred, occluded body parts can be identified, and body parts between people in close proximity of each other can also be disambiguated. However, it cannot be performed with low latency and cannot be applied in an embedded device and used for a UAV.

Once the human pose is detected, under the control of the human pose and referring to the interaction regulation scheme developed in this paper, the UAV would respond and approach the human for further particular commands. However, the UAV's positional motion is coupled with rotary movement, and both of them can be influenced easily. When performing tasks, it is normal for a UAV to encounter wind gust disturbance, which would affect the stability of the whole system. Moreover, to carry out exploration tasks that may be encountered in the future, the UAV is equipped with a 2-DOF robotic arm, which would bring more model uncertainties to the overall system. The

disturbance estimation and attenuation are thus the next problem to overcome. Several similar works have been carried out, such as on disturbance and uncertainty estimation and attenuation (DUEA) strategy, which has been widely used and explored in recent years (Yang et al., 2016). Also, numerous observers have been designed to solve this problem, for example, a disturbance observer (DO) (Zhang et al., 2018; Zhao and Yue, 2018) and extended state observer (ESO) (Shao et al., 2018). Moreover, Mofid and Mobayen (2018) proposes a technique of adaptive sliding mode control (ASMC) for finite-time stabilization of a UAV system with parametric uncertainties. Additionally, a higher-order EDO was applied for attitude stabilization of flexible spacecraft while investigating the effects of different orders on the performance of the EDO (Yan and Wu, 2017). It has been proved that the estimation accuracy can be improved with an increase in the observer order via choosing suitable observer gains. Nevertheless, a higher order of the observer will lead to both high implementation cost and the problem of high gain for observers.

In this paper, an intuitive, natural, end-to-end human-UAV interaction system is built for field exploration assistance. The entire attitude dynamic model of the hexacopter UAV equipped with a robotic arm is presented considering the robotic arm as an element affecting the COG of the vehicle. Moreover, through replacing the backbone network VGG-19 in Cao et al. (2017) by the first twelve layers of MobileNetV2, a real time multi-human pose estimation system, which can be performed with lower latency, maintaining the competitive performance, is built for humans to communicate with the UAV under a proposed interaction regulation. Both target flight direction and distance commands can be transmitted to the UAV easily and naturally. In addition, as a UAV equipped with a robotic arm has more model uncertainty than traditional UAVs and wind gust cannot usually be avoided when carrying out exploration tasks, a composite controller is designed by combining STESO (Shi et al., 2018b) and a back-stepping control method. As most of the disturbances, including wind gust and model uncertainties, are compensated by the feedforward compensator based on STESO, only a small switching gain is required in the controller. Thus, high-accuracy UAV attitude tracking can be realized, and chattering can be alleviated in the presence of several disturbances. Moreover, depth estimation with a binocular camera was developed according to the work of Zhang (2000). The effectiveness of the proposed interaction system and its individual components is demonstrated in several simulations and experiments.

The outline of this work is as follows. Some preliminaries, including quaternion operations and the kinematic and dynamic attitude models of the whole hexacopter UAV are presented in section 2. In section 3, several methods such as human pose estimation, depth estimation, STESO construction, attitude controller, and interaction regulation scheme are formulated. Several simulations and experiments are then given in sections 4 and 5, respectively. Finally, the conclusion is summarized in section 6.

2. PRELIMINARIES

2.1. Notation

The maximal and minimum eigenvalues of matrix H are given by $\lambda_{\max}(H)$ and $\lambda_{\min}(H)$, respectively, and $\|\cdot\|$ represents the 2-norm of a vector or a matrix. Additionally, the operator $S(\cdot)$ denotes a vector $\boldsymbol{\kappa} = \begin{bmatrix} \kappa_1 & \kappa_2 & \kappa_3 \end{bmatrix}^T$ to a skew symmetric matrix as:

$$S(\boldsymbol{\kappa}) = \begin{bmatrix} 0 & -\kappa_3 & \kappa_2 \\ \kappa_3 & 0 & -\kappa_1 \\ -\kappa_2 & \kappa_1 & 0 \end{bmatrix} \quad (1)$$

The sign function can be described as:

$$sign(\kappa) = \begin{cases} \frac{\kappa}{|\kappa|}, & |\kappa| \neq 0 \\ 0, & |\kappa| = 0 \end{cases} \quad (2)$$

2.2. Quaternion Operations

As traditional methods used for representing rotation of the UAV, for instance, the Euler angles, may lead to the singularity problem of trigonometric functions, the unit quaternion $\boldsymbol{q} = \begin{bmatrix} q_0 & \boldsymbol{q_v} \end{bmatrix}^T \in \boldsymbol{R}^4$, $\|\boldsymbol{q}\| = 1$ is utilized in this work Shastry et al. (2018). Several corresponding operations are defined as follows.

The quaternion multiplication:

$$\boldsymbol{q} \otimes \boldsymbol{\sigma} = \begin{bmatrix} q_0\sigma_0 - \boldsymbol{q_v}^T\boldsymbol{\sigma_v} \\ q_0\boldsymbol{\sigma_v} + \sigma_0\boldsymbol{q_v} - \boldsymbol{S}(\boldsymbol{\sigma_v})\boldsymbol{q_v} \end{bmatrix} \quad (3)$$

The relationship between rotation matrix $\boldsymbol{C_A^B}$ and unit quaternion $\boldsymbol{q}$ is described as:

$$\boldsymbol{C_A^B} = (q_0^2 - \boldsymbol{q_v}^T\boldsymbol{q_v})\boldsymbol{I_3} + 2\boldsymbol{q_v}\boldsymbol{q_v}^T + 2q_0\boldsymbol{S}(\boldsymbol{q_v}) \quad (4)$$

The time derivative of Equation (4) is:

$$\dot{\boldsymbol{C}_A^B} = -\boldsymbol{S}(\boldsymbol{\omega})\boldsymbol{C_A^B} \quad (5)$$

where the details of coordinate systems $\boldsymbol{A}$ and $\boldsymbol{B}$ will be given in the next section. Then, the derivative of a quaternion and the quaternion error $\boldsymbol{q_e}$ are given as follows, respectively:

$$\dot{\boldsymbol{q}} = \begin{bmatrix} \dot{q}_0 \\ \dot{\boldsymbol{q}}_v \end{bmatrix} = \frac{1}{2}\boldsymbol{q} \otimes \begin{bmatrix} 0 \\ \boldsymbol{\omega} \end{bmatrix} = \frac{1}{2}\begin{bmatrix} -\boldsymbol{q_v}^T \\ \boldsymbol{S}(\boldsymbol{q_v}) + q_0\boldsymbol{I_3} \end{bmatrix}\boldsymbol{\omega} \quad (6)$$

$$\boldsymbol{q_e} = \boldsymbol{q_d^*} \otimes \boldsymbol{q} \quad (7)$$

where $\boldsymbol{q_d}$ denotes the desired quaternion whose conjugate is represented by $\boldsymbol{q_d^*} = \begin{bmatrix} q_{d0} & -\boldsymbol{q_{dv}} \end{bmatrix}^T$, $\boldsymbol{\omega}$ is the angular velocity of the system.

2.3. Kinematic and Dynamic Models of Hexacopter UAV

As depicted in **Figure 1**, The whole UAV system used for interaction with humans is a hexacopter equipped with a 2-DOF robotic arm. The robotic arm is fixed at the geometric center of the hexacopter. The kinematic and dynamic models of the system are detailed below.
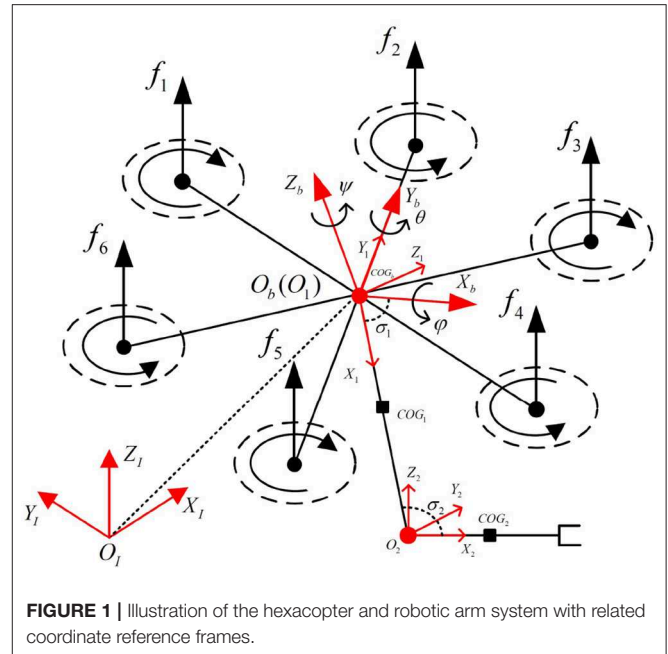
FIGURE 1 | Illustration of the hexacopter and robotic arm system with related coordinate reference frames.

2.3.1. Kinematic Model

The kinematic model of the UAV system can be achieved with several related reference coordinates in **Figure 1**, which are defined as follows:

O_I: world-fixed inertial reference frame

O_b: hexacopter body-fixed reference frame located at the geometric center of the vehicle

O_d: desired reference frame located at the geometric center of the vehicle

O_i: frame fixed to link i in the robotic arm. $i = \{1, 2\}$.

Additionally, several coefficients are used to describe the overall system. $\boldsymbol{\Gamma} = [x, y, z]^T$ represents the absolute position of O_b with reference to O_I. The UAV attitudes are described by Euler angles $\boldsymbol{\Psi} = [\varphi, \theta, \psi]^T$ whose components represent roll, pitch, and yaw angles, respectively. In addition, the absolute linear velocity of the hexacopter with respect to O_b is denoted by $\boldsymbol{V} = [v_x, v_y, v_z]^T$, and $\boldsymbol{\omega} = [\omega_x, \omega_y, \omega_z]^T$ represents the vector of the absolute rotational velocity of the hexacopter with respect to O_b. The relation can be described as:

$$\boldsymbol{\omega} = \boldsymbol{R_r}\dot{\boldsymbol{\Psi}} \quad (8)$$

where

$$\boldsymbol{R_r} = \begin{bmatrix} 1 & 0 & -s\theta \\ 0 & c\varphi & s\varphi c\theta \\ 0 & -s\varphi & c\varphi c\theta \end{bmatrix} \quad (9)$$

where $c(\cdot)$ and $s(\cdot)$, mentioned above, are the abbreviations of $cos(\cdot)$ and $sin(\cdot)$.

2.3.2. Dynamic Model

A traditional UAV with a constant COG at its geometrical center can be described with simple dynamic model equations (Bouabdallah and Siegwart, 2005). However, the

motion of a robotic arm will affect the position of the vehicle. To consider the robotic arm as an element leading to the displacement of the COG from the geometric center of the vehicle, a dynamic attitude model of the whole system is provided in this subsection. Referring to our previous work Jiao et al. (2018), it can be given as:

$$\begin{cases} J_x\dot{\omega}_x = u_1 - (J_z - J_y)\omega_y\omega_z - mc_1 + d_x \\ J_y\dot{\omega}_y = u_2 - (J_x - J_z)\omega_x\omega_z - mc_2 + d_y \\ J_z\dot{\omega}_z = u_3 - (J_y - J_x)\omega_x\omega_y - mc_3 + d_z \end{cases} \quad (10)$$

where

$$\begin{cases} c_1 = y_G(\dot{v}_z - v_x\omega_y + v_y\omega_x) - z_G(\dot{v}_y - v_z\omega_x + v_x\omega_z) \\ c_2 = -x_G(\dot{v}_z - v_x\omega_y + v_y\omega_x) + z_G(\dot{v}_x - v_y\omega_z + v_z\omega_y) \\ c_3 = -y_G(\dot{v}_x - v_y\omega_z + v_z\omega_y) + x_G(\dot{v}_y - v_z\omega_x + v_x\omega_z) \end{cases}$$

$$(11)$$

We describe Equation (11) in a collective form:

$$J\dot{\omega} = u - S(\omega)J\omega - mc + d \quad (12)$$

where vector $J = diag(J_x, J_y, J_z)$ indicates that the inertia matrix is diagonal, and $d = [d_x, d_y, d_z]^T$ denotes the lumped disturbances caused by wind gusts, model uncertainties, etc. The COG of the whole UAV system is described by $C_G = [x_G, y_G, z_G]^T$. m is the total mass of the UAV. Additionally, we define vector $c = [c_1, c_2, c_3]^T$ and vector $u = [u_1, u_2, u_3]^T$, representing the control torque inputs, in which the torques around $x-$, $y-$, and $z-$ generated by the six propellers are represented by u_1, u_2, and u_3, respectively. This has the following expression:

$$u = \Xi f_v \quad (13)$$

where $f_v = [\omega_1^2, \omega_2^2, \omega_3^2, \omega_4^2, \omega_5^2, \omega_6^2]^T$ represents a positive correlation vector with forces generated from the hexacopter motors, in which ω_i denotes the rotor speed of the hexacopter ($i = 1, 2, 3, 4, 5, 6$). In addition, referring to the hexacopter model in **Figure 1**, Ξ can be expressed as follows:

$$\Xi = \begin{bmatrix} \frac{l}{2}\Lambda_T & l\Lambda_T & \frac{l}{2}\Lambda_T & -\frac{l}{2}\Lambda_T & -l\Lambda_T & -\frac{l}{2}\Lambda_T \\ -\frac{\sqrt{3}}{2}l\Lambda_T & 0 & \frac{\sqrt{3}}{2}l\Lambda_T & \frac{\sqrt{3}}{2}l\Lambda_T & 0 & -\frac{\sqrt{3}}{2}l\Lambda_T \\ \Lambda_C & -\Lambda_C & \Lambda_C & -\Lambda_C & \Lambda_C & -\Lambda_C \end{bmatrix}$$

$$(14)$$

where Λ_T and Λ_C denote the thrust and drag coefficients, respectively. Moreover, l represents the distance from each motor to the center of mass of the hexacopter.

3. METHODS

3.1. Human Pose Estimation

Human pose estimation is a prerequisite component of the human-UAV interaction system. It efficiently detects the 2D poses of people in an image. The pose information serves as the coded target within the human-UAV communication, in which each pose form is designed as a special command, guiding the UAV to perform desired tasks.

The challenges of human pose estimation are two-fold. First, under uncertainties, each image may contain multiple people

in various positions and at different scales. Vision-based pose estimation may easily suffer from distraction by irrelevant people, which requires us to design an identification algorithm. It must ignore the non-target candidate people and thus choose the right commander. Second, the above-mentioned commander identification is under the premise that all candidate people can be detected. If we equip each person with a pose detector, the runtime is proportional to the number of people. This would bring significant latency and severely deteriorate the stability of interaction.

To build a time-consuming multi-human pose estimation system, we follow Cao et al. (2017) to employ a bottom-up pose predictor, which means that part locations are first detected and then associated to limbs. Unlike top-down approaches that infer the limb based on each person detection, the bottom-up approach decouples time complexity from the number of people. Specifically, we adopt a two-branch neural network to learn part locations and their associations, respectively. Both of them contribute to the subsequent multi-person parsing process.

The network architecture remains the same as that in Cao et al. (2017), in which an image is taken as input and the connected limbs, i.e., poses, of multiple people are outputs. The raw image first passes through a stack of convolutional layers, generating a set of feature maps. In this stage, we replace VGG-19 (Simonyan and Zisserman, 2014) by the first twelve layers of MobileNetV2 (Sandler et al., 2018) to make it more lightweight, as VGG-19 results in large computational costs and repeatedly employs small-size (3 × 3) convolutional filters to enhance network capacity. In contrast to VGG-19, MoblieNetV2 adopts a novel depthwise separable convolution to reduce actual latency while maintaining competitive performance. The feature maps can be regarded as deep semantic representations of the image, which are then fed into two convolutional branches. The confidence maps and part affinity fields are produced from two branches in parallel. The confidence map predicts the possibility that a particular part occurs at each pixel location, and the part affinity fields measure the confidence of part-to-part association. Finally, the network implements multi-person parsing, which assembles the parts to form the full-body poses of all of the people.

Through this pipeline, multi-human pose estimation can be performed with low latency. The time efficiency is derived not only from the bottom-up inference approach but also from the backbone network used. The bottom-up inference makes run time irrespective of the number of people, allowing the potential for real-time multi-human pose estimation. The selected MobilenetV2 further reduces the number of operations during inference by avoiding large intermediate tensors. To investigate the performance, we train our network on an MPII Multi-person dataset (Andriluka et al., 2014) and test it on our own datasets. During training, the image is resized to (432 × 368). We apply the Adam optimizer (Kingma and Ba, 2014) with default settings ($\varepsilon = 10^{-3}, \beta_1 = 0.9, \beta_2 = 0.999$). The learning rate is set to 0.001, and the batch size is 64. The result for the human image is shown in **Figure 2**. It can be clearly seen that all human poses are correctly detected. Notably, our system achieves a frame-rate of about 6 fps running on an NVIDIA TX2 and, when we adopt the VGG-19 as the backbone, the
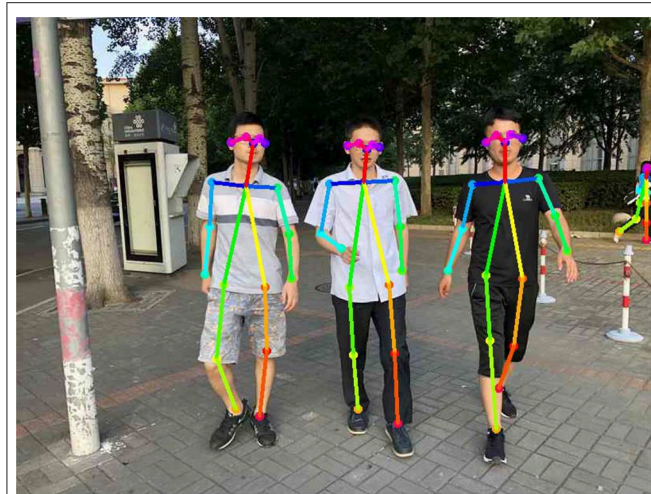
FIGURE 2 | Result of the human pose-estimation system. Written informed consent for publication was obtained from the individuals in this image.

frame-rate drops to about 2 fps. This proves the suitability of MobileNetV2 for mobile applications, especially our human-UAV interaction system.

3.2. Depth Estimation

The depth estimation for the camera installed on the UAV is conducted using a binocular stereo vision ranging method, which is composed of four main parts, namely camera calibration, stereo calibration, stereo rectification, and image matching. The internal and external parameters of the camera are obtained in the camera calibration step, referring to the method of Zhang (2000). Stereo calibration is performed to get the pose and position of one camera with respect to the other. In addition, stereo rectification is used to align image rows between two cameras. The disparity value, which is essential for determining the distance between object and camera, can then be obtained through only searching one row in the image matching step for a match with a point in the other image after the target point is determined. Obviously, this will enhance computational efficiency. Both the stereo rectification and image matching steps are conducted with the use of OpenCV functions. Then, referring to Xuezhi (2014), the depth can be obtained after several works mentioned above.

3.3. Super Twisting Extended State Observer (STESO)

The UAV equipped with a robotic arm has more model uncertainty than a traditional UAV. Moreover, other external disturbances such as wind gusts cannot usually be avoided when carrying out exploration tasks. In this section, all of the disturbances exerted on a UAV are seen as a lumped disturbance, and a STESO is built to estimate it in finite time.

The accelerated velocities $\dot{v}$ and angular velocities ω can be measured by a MEMS accelerometer and gyroscope, respectively, and the lateral velocities can be obtained directly from GPS.

Regarding the dynamics (Equation 12) of the whole UAV system, by importing the feedback linearization method, the original control input can be reformulated as:

$$u = u^* + S(\omega)J\omega + mc \qquad (15)$$

The linearized dynamic model can then be given as:

$$J\dot{\omega} = u^* + d \qquad (16)$$

When building the STESO, it is assumed that each channel is independent, so only one portion is introduced in this subsection and the other two are completely identical. Regarding Equation (16), the one-dimensional dynamics of the UAV used for building the STESO is given as:

$$J_i\dot{\omega}_i = u_i^* + d_i \qquad (17)$$

By importing a new extended state vector $\zeta_i = [\zeta_{i,1}, \zeta_{i,2}]^T$, in which $\zeta_{i,1} = J_i\omega_i$ and $\zeta_{i,2} = d_i, (i = x, y, z)$, the original dynamic model can be constructed as follows:

$$\begin{cases} \dot{\zeta}_{i,1} = u_i^* + \zeta_{i,2} \\ \dot{\zeta}_{i,2} = \chi_i \end{cases} \qquad (18)$$

where χ represents the derivative of d_i and it is assumed that $|\chi| < v^+$, meaning that the lumped disturbance, is bounded.

As the system Equation (18) is observable, the STESO can be designed for this system by introducing a super-twisting algorithm (Yan and Wu, 2019):

$$\begin{cases} \dot{z}_1 = z_2 + u_i^* + \xi_1|e_1|^{\frac{1}{2}}sign(e_1) \\ \dot{z}_2 = \xi_2 sign(e_1) \end{cases} \qquad (19)$$

where z_1 and z_2 represent estimates of $\zeta_{i,1}$ and $\zeta_{i,2}$, respectively. $e_1 = \zeta_{i,1} - \hat{\zeta}_{i,1}$ and $e_2 = \zeta_{i,2} - \hat{\zeta}_{i,2}$ are estimate errors. The whole system estimate errors e_1 and e_2 can be ensured to converge to zero within finite time with appropriate observer gains ξ_1 and ξ_2.

Proof. According to Equations (18) and (19), the error dynamics of the STESO can be obtained as:

$$\begin{cases} \dot{e}_1 = e_2 - \xi_1|e_1|^{\frac{1}{2}}sign(e_1) \\ \dot{e}_2 = \chi - \xi_2 sign(e_1) \end{cases} \qquad (20)$$

Through defining $\delta_i = [\delta_{i,1}{}^T, \delta_{i,2}{}^T]^T$, $\delta_{i,1} = |e_1|^{\frac{1}{2}}sign(e_1)$, $\delta_{i,2} = e_2$, it can be derived that

$$\begin{cases} \dot{\delta}_{i,1} = -\frac{\xi_1}{2}|e_1|^{-1}e_1 + \frac{1}{2}|e_1|^{-\frac{1}{2}}e_2 \\ \dot{\delta}_{i,2} = -\xi_2|e_1|^{-1}e_1 + \chi \end{cases} \qquad (21)$$

Then, we define a positive definite matrix $\eta_1 = \frac{1}{2}\begin{bmatrix} 4\xi_2 + \xi_1{}^2 & -\xi_1 \\ -\xi_1 & 2 \end{bmatrix}$ and introduce the Lyapunov function as:

$$V_i = \delta_i{}^T \eta_1 \delta_i \qquad (22)$$
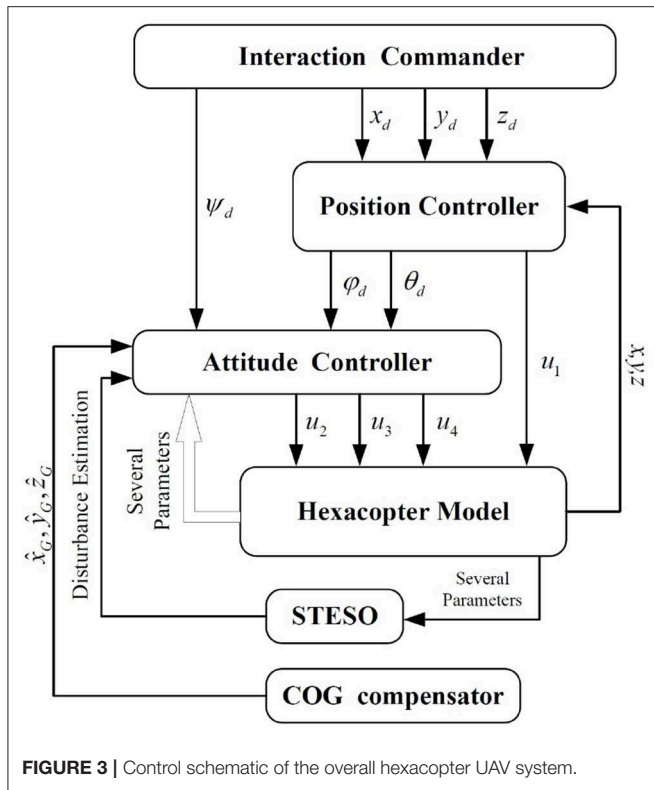
FIGURE 3 | Control schematic of the overall hexacopter UAV system.

We introduce $\boldsymbol{\eta_2} = \frac{\xi_1}{2} \begin{bmatrix} 2\xi_2 + \xi_1{}^2 - \frac{2\nu(\xi_1+1)}{\xi_1} & -\xi_1 \\ -\xi_1 & 1 - \frac{2\nu}{\xi_1} \end{bmatrix}$ and take the time derivative of V_i:

$$
\begin{aligned}
\dot{V}_i &\leq -2\xi_1 |e_1|^{-\frac{1}{2}} [(\xi_1{}^2 + 2\xi_2)\delta_{i,1}{}^2 - 2\xi_1 \delta_{i,1}\delta_{i,2} + \delta_{i,2}{}^2] \\
&\quad + 2\xi_1 |e_1|^{-\frac{1}{2}} (2\delta_{i,1}{}^2 + 2\xi_1{}^{-1}\delta_{i,1}{}^2 + 2\xi_1{}^{-1}\delta_{i,2}{}^2)\nu \\
&= -2\xi_1 |e_1|^{-\frac{1}{2}} \boldsymbol{\delta_i}^T \boldsymbol{\eta_2} \boldsymbol{\delta_i}
\end{aligned}
\tag{23}
$$

It can be found that $\dot{V}_i$ is a negative definite in the case that η_2 is a positive definite. We can then obtain

$$
\begin{cases}
\xi_1 > 2\nu \\
\xi_2 > \frac{\xi_1{}^2}{\xi_1 - 2\nu}\nu + \frac{\xi_1 + 1}{\xi_1}\nu
\end{cases}
\tag{24}
$$

Based on Lyapunov stability theory, we can obtain $|e_1|^{\frac{1}{2}} sign(e_1) \rightarrow 0$ and $e_2 \rightarrow 0$. In this case, the estimate errors e_1, e_2 will converge to zero.

3.4. UAV Controller Approach

The hexacopter, whose rotational motion is coupled with translational motion, is difficult to control to perfection. In **Figure 3**, a control scheme is presented that improves the stability of the system. The control system is cascaded, being composed of two stages, namely the position controller and attitude controller. At the start of the control process, the desired positions (x_d, y_d, z_d) will be sent to the position controller, which will then generate the desired attitudes (φ_d, θ_d) and transmit them

to the attitude controller. The outputs of the attitude controller, which is responsible for guaranteeing that the attitudes track the desired orientations in a finite time, are the desired actuation forces generated by the hexacopter propellers. In addition, a COG compensator is incorporated to work out the real COG and transmit it to the whole control system.

In this section, a traditional PID controller is built for position control. It is only used to generate desired attitudes at translational directions, and the UAV will get to the desired position if the actual attitudes can track the desired orientations in a finite time.

3.4.1. Attitude Control

In order to achieve high-precision attitude tracking in the presence of wind gusts and model uncertainties, some parameters should first be defined. $\boldsymbol{q_d} = [q_{d0}, \boldsymbol{q_{dv}}]^T$ and $\boldsymbol{\omega_d} = [\omega_{dx}, \omega_{dy}, \omega_{dz}]^T$ represent the attitude and desired angular velocities, respectively. We can then obtain the tracking error vector of the angular velocities $\boldsymbol{\omega_e} = [\omega_{ex}, \omega_{ey}, \omega_{ez}]^T$ as:

$$
\boldsymbol{\omega_e} = \boldsymbol{\omega} - \boldsymbol{C_d^b}\boldsymbol{\omega_d}
\tag{25}
$$

We take the time derivative of $\boldsymbol{\omega_e}$ and substitute Equations (5), (12), and (25) into $\dot{\boldsymbol{\omega}}_e$:

$$
\dot{\boldsymbol{\omega}}_e = S(\boldsymbol{\omega_e})\boldsymbol{C_d^b}\boldsymbol{\omega_d} - \boldsymbol{C_d^b}\dot{\boldsymbol{\omega}}_d - J^{-1}S(\boldsymbol{\omega})J\boldsymbol{\omega} + J^{-1}\boldsymbol{u} - mJ^{-1}\boldsymbol{c} + J^{-1}\boldsymbol{d}
\tag{26}
$$

Also, the dynamics of the attitude tracking error can be obtained according to Equations (6), (7), and (25):

$$
\dot{\boldsymbol{q}}_e = \frac{1}{2}\boldsymbol{q_e} \otimes \begin{bmatrix} 0 \\ \boldsymbol{\omega_e} \end{bmatrix} = \frac{1}{2}\begin{bmatrix} -\boldsymbol{q_{ev}^T} \\ S(\boldsymbol{q_{ev}}) + q_{e0}\boldsymbol{I_3} \end{bmatrix}\boldsymbol{\omega_e}
\tag{27}
$$

The STESO-based backstepping controller for the attitude tracking controller is then developed with reference to Shi et al. (2018a). The backstepping is a very good fit for the cascaded structure of the UAV dynamics. To insure that the attitude tracking error $\boldsymbol{q_e}$ converges to zero, we define the Lyapunov function as:

$$
V_{A1} = \boldsymbol{q_{ev}}^T\boldsymbol{q_{ev}} + (1 - q_{e0})^2
\tag{28}
$$

We take time derivative of V_{A1}:

$$
\dot{V}_{A1} = 2\boldsymbol{q_{ev}}^T\dot{\boldsymbol{q}}_{ev} - 2(1 - q_{e0})\dot{q}_{e0} = \boldsymbol{q_{ev}}^T\boldsymbol{\omega_e}
\tag{29}
$$

By introducing a virtual control $\boldsymbol{\omega_{ed}} = -M_1\boldsymbol{q_{ev}}$, in which M_1 is the gain matrix of the controller, when the angular velocity tracking error $\boldsymbol{\omega_e}$ is equal to $\boldsymbol{\omega_{ed}}$, we can obtain:

$$
\dot{V}_{A1} = -\boldsymbol{q_{ev}}^T M_1\boldsymbol{q_{ev}} \leq 0
\tag{30}
$$

$$
\tilde{\boldsymbol{\omega}}_e = \boldsymbol{\omega_e} + M_1\boldsymbol{q_{ev}}
\tag{31}
$$

We then choose the Lyapunov function as:

$$
V_{A2} = V_{A1} + \frac{1}{2}\tilde{\boldsymbol{\omega}}_e^T J\tilde{\boldsymbol{\omega}}_e + V_x + V_y + V_z
\tag{32}
$$

We take the time derivative of V_{A2} according to Equations (25), (30), and (31):

$$\dot{V}_{A2} = q_{ev}^T \tilde{\omega}_e - q_{ev}^T M_1 q_{ev} + \tilde{\omega}_e^T (J\dot{\omega}_e + JM_1\dot{q}_{ev}) + \dot{V}_x$$
$$+ \dot{V}_y + \dot{V}_z = -q_{ev}^T M_1 q_{ev} + \tilde{\omega}_e^T \left(J(S(\omega_e)C_d^b\omega_d - C_d^b\dot{\omega}_d) \right.$$
$$\left. -S(\omega)J\omega + u - mc + d + JM_1\dot{q}_{ev} + q_{ev} \right) + \dot{V}_x + \dot{V}_y + \dot{V}_z$$
(33)

By introducing the control input vector u:

$$u = -J(S(\omega_e)C_d^b\omega_d - C_d^b\dot{\omega}_d) + S(\omega)J\omega + mc - JM_1\dot{q}_{ev}$$
$$-q_{ev} - M_2\tilde{\omega}_e - \hat{d}$$
(34)

$\dot{V}_{A2}$ can be obtained by in substituting u.

$$\dot{V}_{A2} = -q_{ev}^T M_1 q_{ev} - \tilde{\omega}_e^T M_2 \tilde{\omega} + \tilde{\omega}_e^T \tilde{d} + \dot{V}_x + \dot{V}_y + \dot{V}_z$$
$$\leq -\lambda_{\min}(M_1)\|q_{ev}\|^2 - \lambda_{\min}(M_2)\|\tilde{\omega}_e\|^2 + \|\tilde{\omega}_e\| \|\tilde{d}\|$$
$$-\lambda_{\min}(\eta_1)(\|\delta_x\|^2 + \|\delta_y\|^2 + \|\delta_z\|^2)$$
$$\leq -\lambda_{\min}(M_1)\|q_{ev}\|^2 - \lambda_{\min}(M_2)\|\tilde{\omega}_e\|^2 + \|\tilde{\omega}_e\| \|\delta\|$$
$$-\lambda_{\min}(\eta_1)(\|\delta_x\|^2 + \|\delta_y\|^2 + \|\delta_z\|^2)$$
$$\leq -\lambda_{\min}(M_1)\|q_{ev}\|^2 - (\lambda_{\min}(M_2) - \frac{1}{2})\|\tilde{\omega}_e\|^2$$
$$-(\lambda_{\min}(\eta_1) - \frac{1}{2})\|\delta\|^2$$
(35)

where $\left\|\tilde{d}\right\| = \|\delta_2\| \leq \|\delta\|$, $\|\tilde{\omega}_e\| \|\delta\| \leq \frac{1}{2}(\|\tilde{\omega}_e\|^2 + \|\delta\|^2)$, $\|\delta\|^2 = \|\delta_x\|^2 + \|\delta_y\|^2 + \|\delta_z\|^2$, and $\lambda_{\min}(M)$ denotes the minimal eigenvalue of M. Thus, $\dot{V}_{A2} \leq 0$ whenever $\lambda_{\min}(\eta_1), \lambda_{\min}(M_2) \geq \frac{1}{2}$. In that case, it can be concluded that the attitude error q_e, angular velocity tracking error ω_e, and estimation errors $\delta_x, \delta_y, \delta_z$ would be uniformly ultimately bounded and exponentially converge to zero.

3.4.2. COG Compensation System
As shown in **Figure 1**, positional variety in the COG of the vehicle will occur when the UAV conducts tasks that involve the motion of the robotic arm. The dynamic model of the whole system will then be changed during the flight referring to Equation (10). Additionally, the stability of the UAV will be impacted. To overcome this problem, a COG compensation system, which will not be shown here due to the limitations of article length but is detailed in our previous work Jiao et al. (2018), can be implemented. However, the real COG cannot be calculated accurately through this system due to several measuring errors. It will also play a part in the model uncertainties included by the lumped disturbance, which will be estimated by the STESO.

3.5. Interaction Between UAV and Human
3.5.1. Interaction Regulation From Human to UAV
An interaction regulation scheme from human to UAV is developed in this section using the human pose. According to the given interaction regulation, the UAV can be attracted by a distant human by their holding a constant pose, which should

FIGURE 4 | Prototype of human joint.
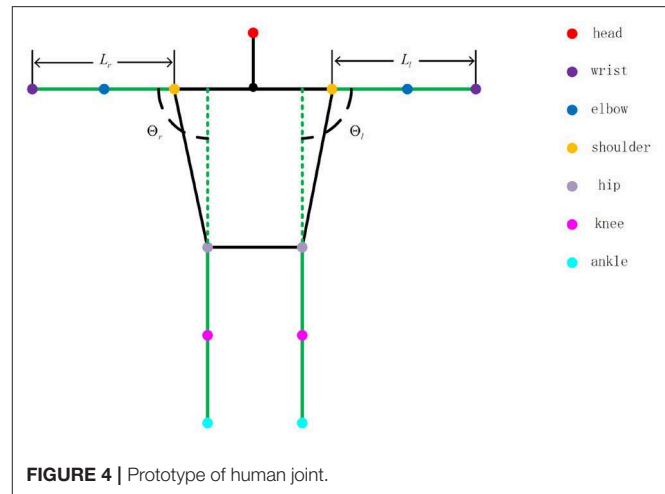
TABLE 1 | Meanings of different coefficient combinations.

Combination	Meaning
$80 < \Theta_l < 100$ and $80 < \Theta_r < 100$	Interaction initiation
$100 < \Theta_l < 150$ and $L_r < L_l$	Flight direction command 1: right front with respect to the human
$30 < \Theta_l < 80$ and $L_r < L_l$	Flight direction command 2: right rear with respect to the human
$100 < \Theta_l < 150$ and $L_l < L_r$	Flight direction command 3: left front with respect to the human
$30 < \Theta_r < 80$ and $L_l < L_r$	Flight direction command 4: left rear with respect to the human
$100 < \Theta_l < 180$ and $100 < \Theta_r < 180$	Flight distance command: based on positions of two wrist joints
$0 < \Theta_l < 30$ and $0 < \Theta_r < 30$	End flag

last more than 5 s, to initiate the interaction. After the interaction initialization is completed, an LED will begin flashing as feedback to the human. Moreover, both target flight direction and distance commands can be communicated to the UAV in a very simple and direct way through human pose changes. As depicted in **Figure 4**, some given straight lines compose a nonobjective human, in which the colorized points, representing the joints of the human body, can be detected and signed by the pose estimation system mentioned in section 3.1. The whole interaction regulation scheme is based on the coefficients (L_l, L_r, Θ_l and Θ_r) given in **Figure 4**. The meanings of different combinations of coefficient values are listed in **Table 1**.

Specifically, the human who is executing search tasks in the field can attract the UAV for search assistance by conducting the interaction initiation action, keeping parallel to the UAV camera, for more than 5 s until the UAV responds by flashing its LED. Moreover, to control the UAV more easily and intuitively, the target command flight direction is just parallel to the human arm, and the target command flight distance is based on the distance between the two wrist joints. As shown in **Figure 5**, the particular flight direction and distance command methods are given, and a nonobjective aerial view of the human, which represents flight
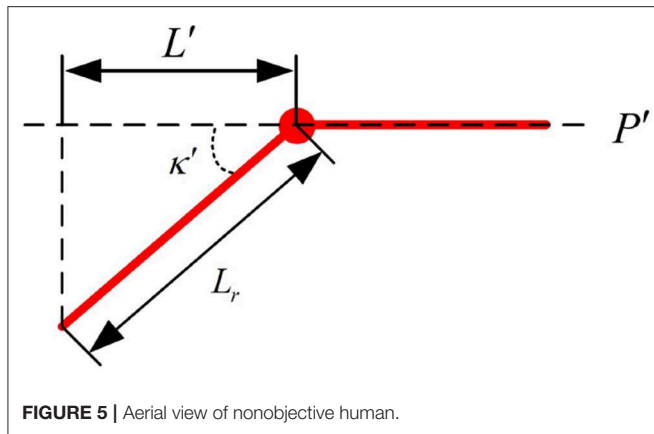
FIGURE 5 | Aerial view of nonobjective human.

Algorithm 1: The Whole Interaction Procedure.

1: Execute human pose estimation system initially and record video from camera;

2: **if** Detect interaction initiation action for more than 5 s **then**

3:    Measure depth information of the detected human and approach him immediately;

4:    Continue to execute human pose estimation system;

5:    **if** interaction initiation **then**

6:      **if** Detect flight direction command **then**

7:        Record target flight direction;

8:        **if** Detect flight distance command **then**

9:          Record target flight distance;

10:          **if** End flag **then**

11:            Execute flight command from human and return to the first line in the procedure;

12:          **end if**

13:        **end if**

14:      **end if**

15:    **end if**

16: **else**

17:    Continue execute human pose estimation system and record the video from camera;

18: **end if**

---

direction command 1 mentioned in **Table 1**, is provided. The two red straight lines are human arms. We can easily obtain the target flight direction with respect to plane P', which is parallel to the UAV camera plane:

$$\kappa' = \arccos \frac{L'}{L_r} \qquad (36)$$

The case with other flight direction commands is similar to that mentioned above. Additionally, the target flight distance command transmitted to the UAV is proportional to the distance between two detected wrist joints. The constant length of a fully stretched human arm, L_l in the picture, represents the unit used as a reference for the distance command. The unit depends on the character of the performed task and would be defined in advance. Moreover, the interaction procedure in the automated exploration task is given in Algorithm 1.

TABLE 2 | Coefficients in the simulation system.

Coefficients	Particulars	Value
m	Mass of the whole UAV system	10.5 kg
J_x	Roll inertia	$4.557 \times 10^{-1} kg \cdot m^2$
J_y	Pitch inertia	$4.557 \times 10^{-1} kg \cdot m^2$
J_z	Yaw inertia	$7.724 \times 10^{-1} kg \cdot m^2$
l	Motor moment arm	0.5 m

3.5.2. Communication From UAV to Human

As shown in **Figure 8**, a vertical column of RGB LEDs, which are used to communicate the state and intents of the UAV to the user as feedback, are fixed on the left undercarriage of the hexacopter. It is controlled by a combination of a pixhawk and an STM32-based board with three colors (red, blue, and green). By changing the color and flicker frequency of the RGB LEDs through the communication regulation formulated in advance, the state and intents of the UAV can be transmitted to the user.

4. SIMULATION RESULTS AND DISCUSSION

To demonstrate the validity and performance of the proposed STESO and corresponding control scheme, several simulations of attitude tracking under external disturbance torque will be conducted using a MATLAB/SIMULINK program with a fixed-sampling time of 1 ms in this section. As a contrast, a traditional second-order ESO is built combined with the proposed attitude controller in the same simulation progress. In addition, we assume that the three-axis components of the external disturbance torques exerted on the UAV are the same and that one of them can be described as:

$$\begin{aligned} d =\ & 0.9 \sin(2.5\pi t - 1) + 1.2 \sin(2\pi t + 2) + 1.95 \sin(0.3\pi t) \\ & + 0.45(0.2\pi t + 6) + 0.15 \sin(0.1\pi) \\ & + 0.75 \sin(0.05\pi - 3.5) + 1.05 \sin(\pi t - 0.9) \\ & + 1.5 \sin(0.01\pi + 1) - 1.185 \end{aligned} \qquad (37)$$

Moreover, the dynamic model built in section 2.3 is taken as the basis of the simulation of the proposed observer and controllers. The simulation parameters, which are verified to be very close to the reality of the single multi-copter and are listed in **Table 2**, are generated by the online toolbox of Quan (2018). Although this is a list of coefficients for a single multi-copter without a robotic arm, it is also useful in our simulation system, as the rest of the model uncertainty can also be included in the lumped disturbance and estimated by the proposed observer. Additionally, we choose the BSC gains as $M_1 = 10I_3$, $M_2 = 3I_3$, the STESO gains as $\xi_{1,i} = 28$, $\xi_{2,i} = 58$, and the traditional second-order ESO gains as $L = [\ 35\ \ 380\ \ 600\ ]^T$.

It can be seen from **Figure 3** that an effective attitude controller is the foundation of UAV motion and needs to work well during a UAV exploration task with unknown external disturbances. As shown in **Figure 6**, several attitude tracking

FIGURE 6 | Simulation results of roll, pitch, and yaw angle-tracking with unknown external torque disturbance. Without observer (Up), traditional ESO (Middle), and STESO (Down).
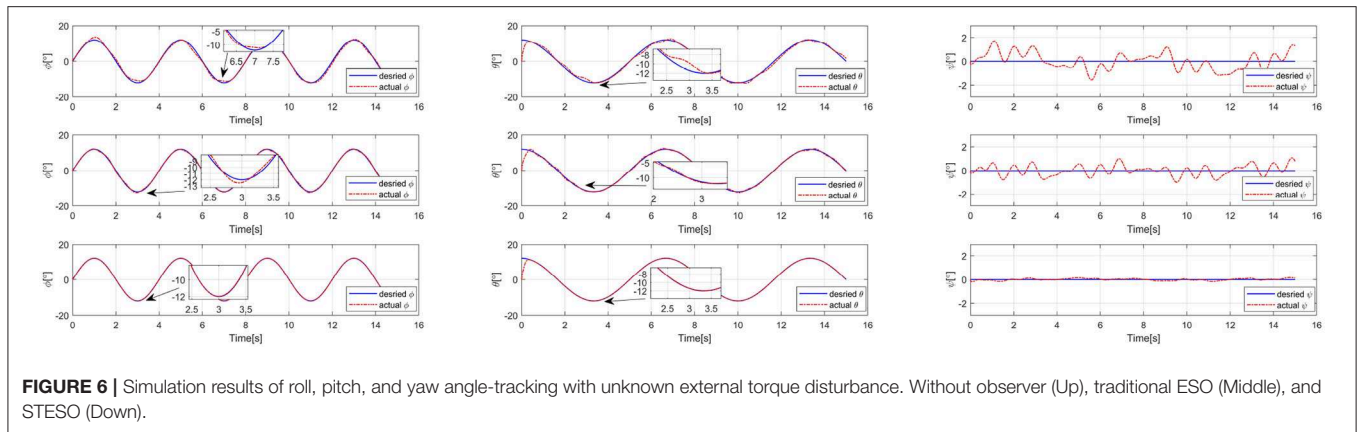
simulations have been conducted based on the proposed back-stepping controller with the STESO and traditional ESO. The desired attitude references are given as:

$$\mathbf{\Theta}_d = [\,12\sin(0.5\pi t) \quad 12\cos(0.3\pi t) \quad 0\,]^T \text{ deg} \qquad (38)$$

From these figures, we can easily determine that the tracking errors are largest in all of the channels (roll, pitch, and yaw) without any observers. The tracking trajectories are influenced seriously. Moreover, we also find that it is obviously improved with observers to estimate and then attenuate the disturbance directly, even though the disturbance is not estimated completely. Compared to being equipped with the traditional ESO, the tracking error is also further reduced by using the STESO. Further, the disturbance estimate errors of the STESO and traditional ESO in all channels are shown in **Figure 7**, showing that the STESO could make a better estimation than the traditional ESO. Thus the UAV can attain better attitude tracking performance under the control of the proposed controller with an STESO.

5. EXPERIMENTAL RESULTS AND DISCUSSION

This section details several experiments, including hovering with wind gusts and a synthetic interaction experiment between humans and a UAV, that were conducted in a playground to validate all the above-mentioned theories.

5.1. Hardware Platform

Figure 8 shows the UAV platform suitable for our interaction system that was constructed. It is a hexacopter with a 143-cm tip-to-tip wingspan, six 17-inch propellers, a height of 58 cm, and a total mass of 10.5 kg including the robotic arm, which is fixed under the vehicle. Each rotor offers lift force of up to 4.0 kg, which is enough for the whole system. In addition, Open-source PIXHAWK hardware (Meier, 2012), which includes an STM32 processor and two sets of IMU sensors, is fastened to the top of the UAV and is used for sensor data integration, attitude computation, mode switching, state assistant feedback, controller and STESO operation, emergency security protection,

FIGURE 7 | Disturbances estimate errors ($\tilde{d}_x$, $\tilde{d}_y$, $\tilde{d}_z$) of the STESO and traditional ESO.
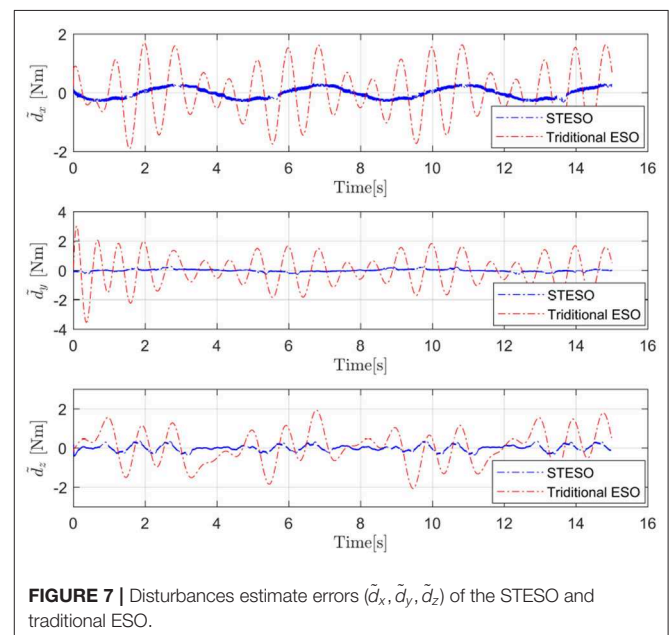
etc. Moreover, an NVIDIA TX2 equipped with six CPU cores and 256 CUDA cores is utilized in the interaction system in which the human pose estimation and depth computation tasks are loaded. A binocular stereo camera, which offers 720P video transmission of up to 60 fps, is placed at the front of the vehicle to obtain the three-dimensional position of the target with respect to the UAV. Additionally, to ensure the safety of the experimental partner during close-range interaction, a high-precision GPS is utilized to supply accurate information on the absolute and relative position of the vehicle, which can also enable stable UAV hovering. Moreover, the vehicle uses HRI-LEDs to communicate its state and intents to the user, and a compass is placed at the highest point of the UAV to prevent electromagnetic interference.

5.2. Hovering With Wind Gusts

To demonstrate the performance of the developed method for a UAV subject to lumped disturbances including wind gusts and
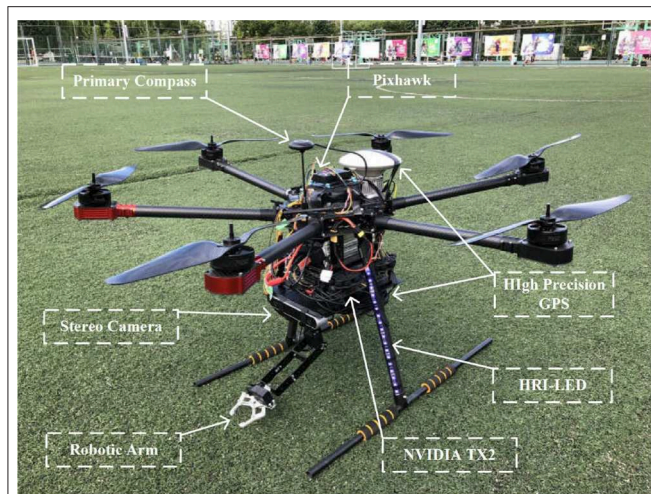
FIGURE 8 | Prototype of the proposed interactive UAV.

FIGURE 10 | Interaction between human and UAV. Written informed consent for publication was obtained from the individuals in this image.
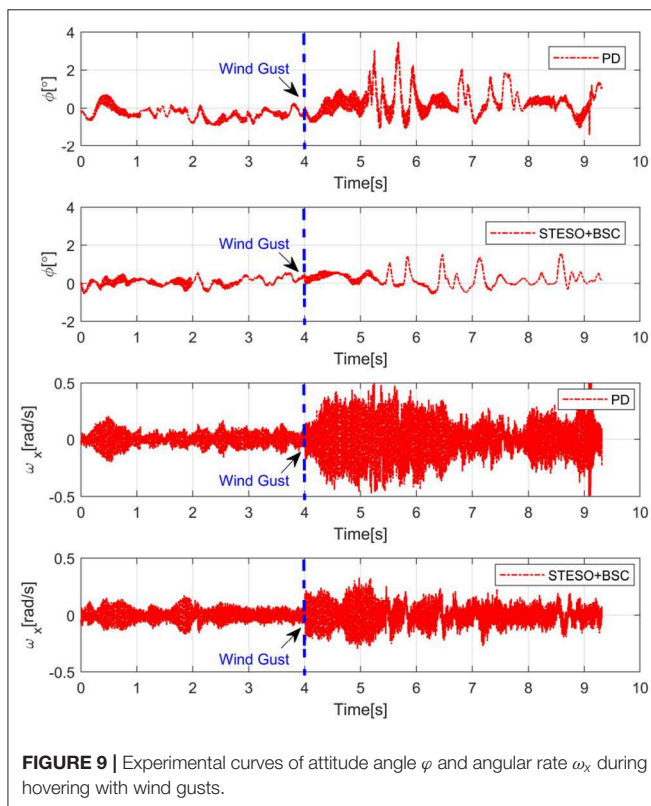
FIGURE 9 | Experimental curves of attitude angle φ and angular rate ω_x during hovering with wind gusts.

FIGURE 11 | Interaction initiation. Written informed consent for publication was obtained from the individuals in this image.

the attitude φ under PD and STESO-BSC are no more than $4°$ and $2°$, respectively. Additionally, under control of PD, the UAV has more drastic chattering with angular velocity varying more quickly.

5.3. Human-UAV Interaction

As shown in **Figure 10**, in this section, an automated exploration task with the proposed interaction system was conducted in a playground. It includes interaction initiation and particular commands, which contain UAV flight target direction and distance, communicated from human to UAV.

5.3.1. Interaction Initiation

As shown in **Figure 11**, the interaction initiation action of a human was detected by the UAV at a distance in the playground. Meanwhile, another person who walked around served as a

model uncertainties, a hovering experiment was conducted with wind disturbance generated by several electrical fans. We set the BSC gains at $M_1 = diag(10, 10, 4)$ and $M_2 = diag(0.2, 0.2, 0.28)$ and the STESO gains at $\xi_{1,i} = 1.2$, $\xi_{2,i} = 0.3$. The results for attitude angle φ and angular velocity ω_x under STESO-BSC and a traditional PD controller can be found in **Figure 9**. It can be observed that the chattering is markedly reduced under STESO-BSC compared to PD. In particular, the peak values of

FIGURE 12 | Target direction and distance communication to the UAV. (1) Beginning. (2) Target direction command. (3) Target distance command. (4) End. Written informed consent for publication was obtained from the individual in this image.
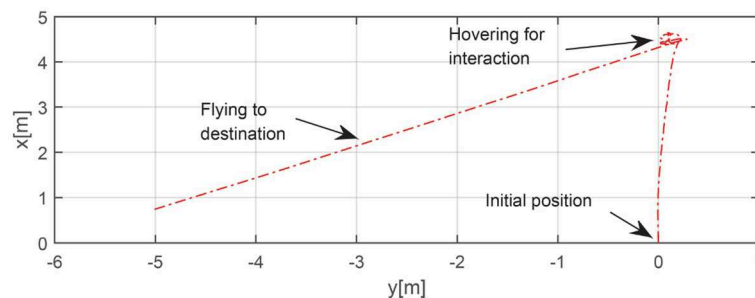
FIGURE 13 | Trajectory in automated exploration task.

disturbance term during the whole interaction initiation process. It had been verified that the UAV can discriminate these poses from human walking and other human motions. All the key joints of the human were obtained from the human pose estimation system. The average inference time per image is about 0.167 s. After the interaction initiation process was finished, the UAV approached human, while flashing its light as feedback, for further command information.

5.3.2. Automated Exploration Task

After the interaction initiation process was completed, the human received the UAV's feedback information and was ready to give the next command to the UAV. The specific steps of close interaction can be seen in **Figure 12**, in which parts (2) and (3) represent direction and distance commands, respectively, as outlined in section 3.5.1. According to the positions of the human joints and Equation (36), the target direction with respect to the camera plane was determined to be 36.5°. Meanwhile, using the reference unit for the distance command set in this experiment, 10 m, the final target distance was determined to be 17.4 m. The flight trajectory was recorded and is shown in **Figure 13**. The actual direction angle and distance are 35.4° and 16.9 m, respectively, the errors of which are small enough for field exploration. However, owing to the limitations of figure space, the trajectory to the destination is shown only partially. We could conclude from the experiment that the proposed interaction system is qualified to complete the field exploration task. However, through the whole experiment, we also found that

the process of the interaction between UAV and human was not quick enough. As the frame-rate of pose estimation is still limited in spite of its improvement through our work, the human has to wait for a while for the response from the UAV at every step of the interaction, which will influence the interactive efficiency and experience. More attention should thus be paid to developing this state-of-art interaction technique in the future.

6. CONCLUSION

In this study, an intuitive end-to-end human-UAV interaction system, in which a UAV can be controlled to fly to a corresponding direction and distance by human poses, was built to assist in field exploration. Moreover, a real time multi-human pose estimation system, which performs with low latency while maintaining competitive performance, was built with which a human can communicate with the UAV under a proposed interaction regulation scheme. By introducing the super-twisting algorithm, an STESO was constructed and applied to the UAV attitude control system to estimate and attenuate complex disturbances, such as wind gusts, model uncertainties, etc. Based on the STESO, a back-stepping attitude controller was built that was proved through several simulations and experiments to have a better performance than a back-stepping controller with a traditional ESO. Finally, an integrated human-UAV interaction experiment was conducted in which the effectiveness of the whole system and its individual components were demonstrated.

DATA AVAILABILITY STATEMENT

The datasets generated for this study are available on request to the corresponding author.

ETHICS STATEMENT

All procedures performed in studies involving human participants were in accordance with the ethical standards of the institutional and/or national research committee and with the 1964 Helsinki Declaration and its later amendments or comparable ethical standards. Informed consent was obtained from all individual participants involved in the study.

AUTHOR CONTRIBUTIONS

RJ conceived and designed the STESO and corresponding back-stepping attitude control algorithm. RJ, ZW, and RC built the human pose estimation system. YR constructed the depth estimation system. MD and WC assisted in the manuscript writing.

FUNDING

REFERENCES

Andriluka, M., Pishchulin, L., Gehler, P., and Schiele, B. (2014). "2d human pose estimation: new benchmark and state of the art analysis," in *Proceedings of the IEEE Conference on computer Vision and Pattern Recognition* (Columbus, OH: IEEE), 3686–3693.

Bin, F., Liu, S. F., Liu, H., and Chunfang, L. (2018). 3d human gesture capturing and recognition by the immu-based data glove. *Neurocomputing* 277, 198–207. doi: 10.1016/j.neucom.2017.02.101

Bouabdallah, S., and Siegwart, R. (2005). "Backstepping and sliding-mode techniques applied to an indoor micro quadrotor," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Barcelona: IEEE), 2247–2252.

Cao, Z., Simon, T., Wei, S.-E., and Sheikh, Y. (2017). "Realtime multi-person 2d pose estimation using part affinity fields," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (Honolulu, HI: IEEE), 7291–7299.

Fang, B., Wei, X., Sun, F., Huang, H., Yu, Y., Liu, H., et al. (2019). Skill learning for human-robot interaction using wearable device. *Tsinghua Sci. Technol.* 24, 654–662. doi: 10.26599/TST.2018.9010096

Jiao, R., Dong, M., Ding, R., and Chou, W. (2018). "Control of quadrotor equipped with a two dof robotic arm," in *3rd International Conference on Advanced Robotics and Mechatronics (ICARM)* (Singapore: IEEE), 437–442.

Kingma, D. P., and Ba, J. (2014). Adam: a method for stochastic optimization. *preprint arXiv:1412.6980*.

Kolling, A., Nunnally, S., and Lewis, M. (2012). "Towards human control of robot swarms," in *Proceedings of the Seventh Annual ACM/IEEE International Conference on Human-Robot Interaction* (Boston, MA: ACM), 89–96.

Lv, X. Z., Wang, M. T., Qi, Y. F., Zhao, X. M., and Dong, H. (2014). Research on ranging method based on binocular stereo vision. *Adv. Mater. Res.* 945, 2075–2081. doi: 10.4028/www.scientific.net/AMR.945-949.2075

McLurkin, J., Smith, J., Frankel, J., Sotkowitz, D., Blau, D., and Schmidt, B. (2006). "Speaking swarmish: human-robot interface design for large swarms of autonomous mobile robots," in *AAAI Spring Symposium* (Palo Alto, CA), 72.

Meier, L., Tanskanen, P., Heng, L., Lee, G. H., Fraundorfer, F., and Pollefeys, M. (2012). Pixhawk: a micro aerial vehicle design for autonomous flight using onboard computer vision. *Auton. Robots* 33, 21–39. doi: 10.1007/s10514-012-9281-4

Mofid, O., and Mobayen, S. (2018). Adaptive sliding mode control for finite-time stability of quad-rotor uavs with parametric uncertainties. *ISA Trans.* 72, 1–14. doi: 10.1016/j.isatra.2017.11.010

Monajjemi, V. M., Wawerla, J., Vaughan, R., and Mori, G. (2013). "HRI in the sky: creating and commanding teams of uavs with a vision-mediated gestural interface," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Tokyo: IEEE), 617–623.

Pishchulin, L., Insafutdinov, E., Tang, S., Andres, B., Andriluka, M., Gehler, P., et al. (2016). "Deepcut: joint subset partition and labeling for multi person pose estimation," in *Proceedings of The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (Seattle, WA: IEEE), 4929–4937.

Pourmehr, S., Monajjemi, V., Wawerla, J., Vaughan, R., and Mori, G. (2013a). "A robust integrated system for selecting and commanding multiple mobile robots," in *IEEE International Conference on Robotics and Automation (ICRA)* (Karlsruhe: IEEE), 2874–2879.

Pourmehr, S., Monajjemi, V. M., Vaughan, R., and Mori, G. (2013b). "'you two! take off!': creating, modifying and commanding groups of robots using face engagement and indirect speech in voice commands," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Tokyo: IEEE), 137–142.

Quan, Q. (2018). *Flight Performance Evaluation of UAVs*. Available online at: http://flyeval.com/

Ran, Y., Weiss, I., Zheng, Q., and Davis, L. S. (2007). Pedestrian detection via periodic motion analysis. *Int. J. Comput. Vis.* 71, 143–160. doi: 10.1007/s11263-006-8575-4

Rodriguez, D. P., Maza, I., Caballero, F., and Scarlatti, D. (2013). A ground control station for a multi-UAV surveillance system. *J. Intell. Robot. Syst.* 69, 119–130. doi: 10.1007/s10846-012-9759-5

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., and Chen, L.-C. (2018). "Mobilenetv2: inverted residuals and linear bottlenecks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (Salt Lake City, UT: IEEE), 4510–4520. doi: 10.1109/CVPR.2018.00474

Shao, X., Liu, J., Cao, H., Shen, C., and Wang, H. (2018). Robust dynamic surface trajectory tracking control for a quadrotor uav via extended state observer. *Int. J. Robust Nonlin. Control* 28, 2700–2719. doi: 10.1002/rnc.4044

Shastry, A. K., Bhargavapuri, M. T., Kothari, M., and Sahoo, S. R. (2018). "Quaternion based adaptive control for package delivery using variable-pitch quadrotors," in *Indian Control Conference (ICC)* (Kanpur: IEEE), 340–345.

Shi, D., Wu, Z., and Chou, W. (2018a). Harmonic extended state observer based anti-swing attitude control for quadrotor with slung load. *Electronics* 7:83. doi: 10.3390/electronics7060083

Shi, D., Wu, Z., and Chou, W. (2018b). Super-twisting extended state observer and sliding mode controller for quadrotor uav attitude system in presence of wind gust and actuator faults. *Electronics* 7:128. doi: 10.3390/electronics7080128

Simonyan, K., and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv: 1409.1556*.

Wu, S., Oreifej, O., and Shah, M. (2011). "Action recognition in videos acquired by a moving camera using motion decomposition of lagrangian particle trajectories," in *IEEE International Conference on Computer Vision (ICCV)* (Barcelona: IEEE), 1419–1426.

Yan, R., and Wu, Z. (2017). Attitude stabilization of flexible spacecrafts via extended disturbance observer based controller. *Acta Astronaut.* 133, 73–80. doi: 10.1016/j.actaastro.2017.01.004

Yan, R., and Wu, Z. (2019). Super-twisting disturbance observer-based finite-time attitude stabilization of flexible spacecraft subject to complex disturbances. *J. Vibrat. Control* 25, 1008–1018. doi: 10.1177/1077546318808882

Yang, J., Chen, W.-H., Li, S., Guo, L., and Yan, Y. (2016). Disturbance/uncertainty estimation and attenuation techniques in pmsm drives-a survey. *IEEE Trans. Indust. Electr.* 64, 3273–3285. doi: 10.1109/TIE.2016.2583412

Zhang, Z., Wang, F., Guo, Y., and Hua, C. (2018). Multivariable sliding mode backstepping controller design for quadrotor uav based on disturbance observer. *Sci. China Informat. Sci.* 61, 3273–3285. doi: 10.1007/s11432-017-9434-7

Zhang, Z. E. A. (2000). A flexible new technique for camera calibration. *IEEE Trans. Patt. Anal. Mach. Intell.* 22, 1330–1334. doi: 10.1109/34.888718

Zhao, B., and Yue, D. (2018). "Disturbance observer based nonlinear robust attitude tracking controller for a hexarotor UAV," in *37th Chinese Control Conference (CCC)* (Wuhan: IEEE), 9996–10001. doi: 10.23919/ChiCC.2018.8483005

Design and Performance Evaluation of a Novel Wearable Parallel Mechanism for Ankle Rehabilitation

Shiping Zuo[1], Jianfeng Li[1], Mingjie Dong[1]*, Xiaodong Zhou[2], Wenpei Fan[1] and Yuan Kong[1]

[1] College of Mechanical Engineering and Applied Electronics Technology, Beijing University of Technology, Beijing, China,
[2] Beijing Institute of Control Engineering, Beijing, China

Repetitive and intensive physiotherapy is indispensable to patients with ankle disabilities. Increasingly robot-assisted technology has been employed in the treatment to reduce the burden of the therapists and the related costs of the patients. This paper proposes a configuration of a wearable parallel mechanism to supplement the equipment selection for ankle rehabilitation. The kinematic analysis, i.e., the inverse position solution and Jacobian matrices, is elaborated. Several performance indices, including the reachable workspace index, motion isotropy index, force transfer index, and maximum torque index, are developed based on the derived kinematic solution. Moreover, according to the proposed kinematic configuration and wearable design concept, the mechanical structure that contains a basic machine-drive system and a multi-model position/force data collection system is designed in detail. Finally, the results of the performance evaluation indicate that the wearable parallel robot possesses sufficient motion isotropy, high force transfer performance, and large maximum torque performance within a large workspace that can cover all possible range of motion of human ankle complex, and is suitable for ankle rehabilitation.

Keywords: ankle rehabilitation, parallel robot, mechanical design, performance indices, performance evaluation

INTRODUCTION

As the population ages, increasingly more individuals experience ankle disabilities caused by stroke and cerebral palsy, which may lead to lack of mobilization, irregular pain of body, insufficient capacity to support weight, and chronic joint instability. During the conventional manually physiotherapy, human ankle complex (HAC) is moved by a physical therapist with its range of motion (ROM). However, it possesses many limitations such as, the duration inconsistency and frequency indetermination of the treatment procedures, the physical demand, and experience requirement of the therapist, and the subjective evaluation of the therapeutic results (Meng et al., 2015; Hussain et al., 2017). In view of this situation, to provide high-quality rehabilitation treatment with repetitive sessions, quantitative measurements, scientifical therapy, and systematic operation, robot-assisted rehabilitation has become a field that receives more, and more research attention. To date, various ankle rehabilitation devices have been introduced based on different concepts that can be mainly divided into two categories: ankle exoskeleton and parallel platform-based robots.

Focusing on walking gait treatment on treadmill or over-ground, ankle exoskeletons are wearable with mechanical parts attaching to the human limb. A typical instance is the active ankle-foot orthosis proposed by Blaya and Herr (2004), by employing series elastic actuator (SEA),

rotary potentiometer, and ground reaction force sensors, a gait pathology known as drop-foot can be treated via variable-impedance control. Similarly, exoskeletons (Kim et al., 2011; Zhu et al., 2011; Lopez et al., 2013; Meijneke et al., 2014; Witte et al., 2015; Dijk et al., 2017; Erdogan et al., 2017) were also developed with the concept of SEA to provide push-off assistance. Moreover, robotic tendons, i.e., a DC motor in series with a spring, were used in ankle-foot orthoses (Hollander et al., 2006; Boehler et al., 2008; Oymagil et al., 2008; Ward et al., 2011) to provide sufficient energy and peak power saving for systems; pneumatic muscle actuators (PMA), due to their intrinsically compliant and high power/weight ratio, were also widely selected as the actuation technology of the ankle exoskeletons (Ferris et al., 2006; Gordon et al., 2006; Kinnaird and Ferris, 2009; Sawicki and Ferris, 2009; Park et al., 2014). Additionally, directly aligning several types of actuators, including rotating actuator assembly (Ren et al., 2017), servo motor (Yoshizawa, 2010; Yao et al., 2018) and bidirectional pneumatic actuator (Shorter et al., 2011), to the joint axis is another option for researchers.

The aforementioned exoskeletons possess one degree of freedom (1-DOF) for assisting plantarflexion/dorsiflexion (PL/DO) motion. By applying parallel mechanism-based design, more DOFs can be performed by exoskeletons. A well-known example is the Anklebot proposed by Roy et al. (2009). Two linear actuators were arranged in parallel to aid recovery of PL/DO and inversion/eversion (IN/EV), while the adduction/ abduction (AD/AB) can be achieved via the rotation of the leg. Subsequently, a scaled down version called pediAnklebot (Michmizos et al., 2015) is developed for pediatric rehabilitation. Fan and Yin (2009) presented an ankle exoskeleton with a 3-RPS (revolute-prismatic-spherical) parallel mechanism as the main mechanical structure in cooperation with an electromyographic-based neuro-fuzzy controller. A reconfigurable ankle exoskeleton (Erdogan et al., 2017) was proposed for multiple phases of treatment, in which the 3-RPS structure and the 3-UPS (universal-prismatic-spherical) structure can be interconverted via lockable joints. Stewart platform (Takemura et al., 2012; Nomura et al., 2015) was also utilized for ankle exoskeletons.

In the early stage of rehabilitation, the movements of ankle are weak and stiff due to the muscle atrophy or the loss of physiological muscles synergies. Thus, before performing gait treatment using wearable exoskeleton, a parallel platform-based robot, due to the features of superior adaptability, excellent accuracy and high stiffness, is a more suitable option to ensure reliability and safety in the process of rehabilitation treatment. Moreover, by fixing the foot on the platform, three rotational DOFs (i.e., PL/DO, IN/EV, and AD/AB) of the HAC can be all provided treatment.

Girone et al. (1999) proposed a pneumatically actuated ankle rehabilitation robot (Rutgers Ankle) with a force feedback system. A virtual reality environment has been developed to make rehabilitation more effective and enjoyable. By employing passive central struts in the mechanical structure to determine the number of DOFs and increase the payload capability, lower-mobility parallel robots (Dai and Zhao, 2004; Liu et al., 2006; Saglia et al., 2009) have been proposed for ankle rehabilitation with their DOFs better matching with that of the HAC.

Tsoi et al. (2009) replaced the central strut with the lower-limb of the patient (i.e., the HAC is directly adopted to constrain the motion of the platform). In comparison of the aforementioned platform-based robot, this robot realizes an aligned rotation center between the platform and the HAC in the process of rehabilitation. To avoid the safety issue caused by excessive load in this method, physical rotation axes (i.e., kinematic constraint mechanisms) were specially designed. Specifically, Jamwal et al. (2014) proposed a compliant parallel robot by arranging four PMAs parallel to the shank of the patient. Three bearings were setting into the platform as kinematic constraint. Thanks to the inherent muscle-like behavior, compliant motions can be achieved during different treatment modes with the help of a fuzzy logic controller. Analogous to Jamwal et al. (2014), in Zhang et al. (2017), The University of Auckland developed the other compliant robot powered by four PMAs that arranged in a tilted manner. A three-linkage serial mechanism was set as the kinematic constraint of this robot, and the connection points (i.e., spherical joints) can be adjusted along certain directions to achieve reconfigurable workspace and torque capacity. By selecting two types of identical active branches, i.e., 3-UPS structure and 3-RUS (revolute-universal-spherical), to produce obliquity of the platform and using serial equivalent spherical mechanisms to satisfy all 3-DOF rotational ankle rehabilitation, actuated parallel mechanisms introduced by Wang et al. (2013, 2015) are another typical instance.

Notably, the arrangement of the physical rotation axes has become an effective method for a parallel platform-based ankle rehabilitation robot to actualize the required treatment action, realize aligned rotational center and ensure primary safety. Redundant actuation technology, despite having received widespread application, may present a complicated structure and control scheme, and then increase the manufacturing and operation cost. Moreover, a totally relaxed lower-limb may prevent the HAC from fully stretching into the extreme position, which limits the improvement and functional recovery of muscle strength. Meanwhile, repeatedly changes in sitting posture caused by a loose shank may lead to re-injury to the patient. This paper put forth a novel parallel robot for ankle rehabilitation with a wearable design concept to provide maximum safety protection. A simple configuration is adopted to realize actuator non-redundancy and reduce the relatively cost.

The remainder of this paper is organized as follows: The HAC anatomy and configuration design are presented in section HAC Anatomy and Configuration Design. In section Kinematic Analysis, the kinematic analysis, including inverse position solution and velocity Jacobian matrices, are derived, based on which several performance indices are defined in section Performance Indices. Section Mechanical Design describes the mechanical design in detail. Section Performance Evaluation analyzes the performance including reachable workspace, motion isotropy, force transfer performance and maximum torque performance. Finally, we discuss the main findings and draw the main conclusions of the study.

MATERIALS AND METHODS

HAC Anatomy and Configuration Design

Considering HAC anatomy in the design process of an ankle rehabilitation robot is a basic guarantee to ensure the comfort and safety of patients during rehabilitation. Thus, it is necessary to carry out the anatomical analysis before determining the configuration of the robot. As one of the most complicated joint in the human body, the HAC (**Figure 1A**) contains two anatomically separate joints, namely, the ankle joint and the subtalar joint (Dai and Zhao, 2004; Khalid et al., 2015). Specifically, the ankle joint consisting of the tibia, fibula and talus, is located above the subtalar joint which is formed by the talus inferiorly and the calcaneus superiorly (Dai and Zhao, 2004). Moreover, three rotational motions, i.e., PL/DO, IN/EV, and AD/AB, resulted from the interaction between the articulating joint surfaces and the constrained ligament constitute the basic motion form of the HAC (Isman and Inman, 1969). The rotation axis of the ankle joint (i.e., PL/DO) passes through the tips of the medial and lateral malleolus (**Figure 1B**), and the orientation of IN/EV (i.e., the rotation axis of the subtalar joint) is approximated by the line between the superior point of the navicular and the posterolateral point of the calcaneus (**Figure 1C**; Dul and Johnson, 1985; Dettwyler et al., 2004). The combined motion of the ankle joint and the subtalar joint, as well as the rotation between the tibia and fibula contribute to the AD/AB (Khalid et al., 2015). In biology, the aforementioned skewed rotation axes produce rotational motion in all three orthogonal planes (i.e., sagittal, coronal, and transverse planes; Feuerbach et al., 1994). Thus, when considering the kinematic model of the HAC from the perspective of mechanism, the two separate subjoints can be

simply regarded as a 3-DOF spherical joint in a combined manner (**Figure 1D**).

According to the kinematic model of the HAC, a serial constraint branch (**Figure 2A**) with a three axes-intersected revolute joints (R_1, R_2, and R_3) structure is selected as an equivalent spherical joint to imitate the 3-DOF rotational motion of the HAC and determine the rotation center of the robot. By using this constraint branch, the ankle rehabilitation treatment can be performed under a human-robot compatible situation with fixed rotation center and precise DOFs. Moreover, two identical non-constraint rods with UPS structure are selected as the kinematic branches. Based on the aforementioned consideration, a 2-UPS/RRR parallel mechanism is proposed as the basic configuration of the ankle rehabilitation robot. Joints P_1, P_2, and R_1 are active, whereas all others are passive. For patients with ankle disabilities, the movement of the HAC becomes weaker and stiffer, and even with ankle spasticity/contracture (Zhou et al., 2014). Thus, to protect the already fragile ankle from secondary injury, the position of the shank and the HAC should remain stationary with respect to the foot. Additionally, a follow-up shank makes it difficult to perform treatment at the extreme position of the HAC, thus affecting the full recovery of function. By inverting the 2-UPS/RRR parallel mechanism (i.e., the base is positioned above the moving platform), and inserting the shank of the patient into the mechanism as a part of the base via an accessory wearable binding mechanism, decoupled foot-shank motion (i.e., the shank will not move with the foot) and the maximum safety guarantee can be achieved during different treatment modes.

The formed human-robot system consisting of the HAC and the wearable parallel robot is presented in **Figure 2B**. Analogous

FIGURE 1 | The HAC anatomy. **(A)** The structure of the HAC. **(B)** Rotation axis of PL/DO. **(C)** Rotation axis of IN/EV. **(D)** Motion form of the HAC.

FIGURE 2 | **(A)** Constraint branch. **(B)** Human-robot system.
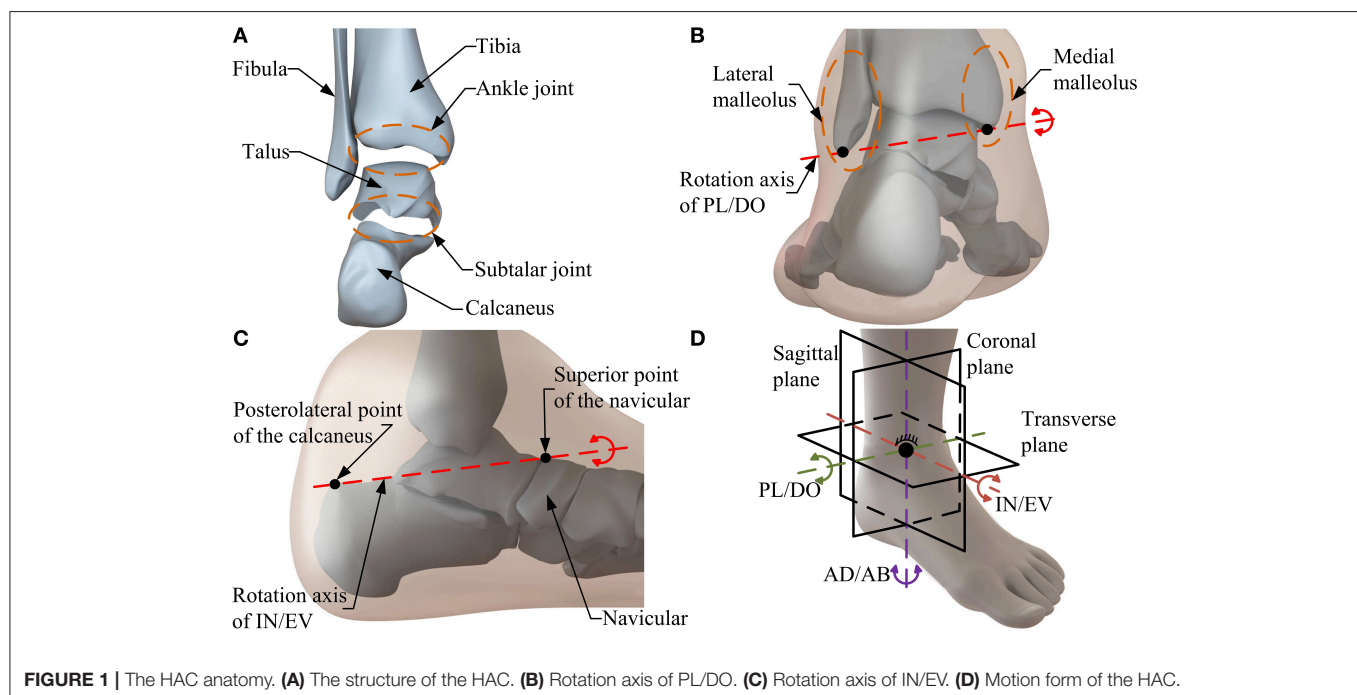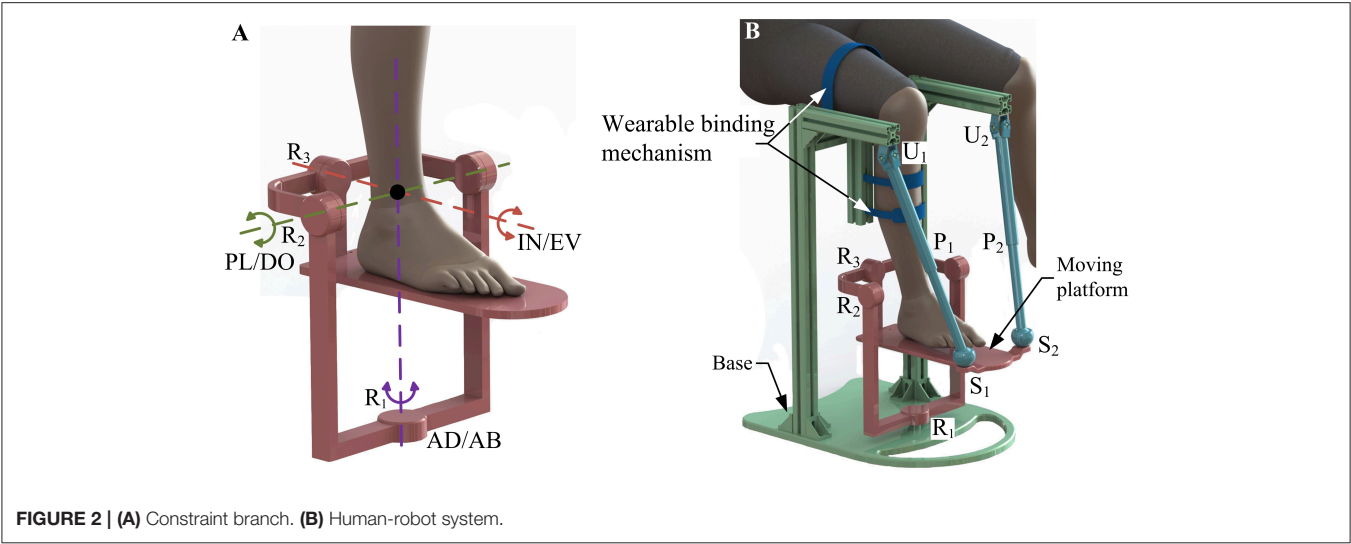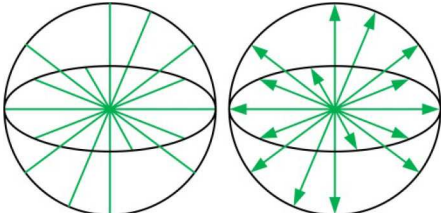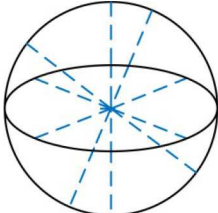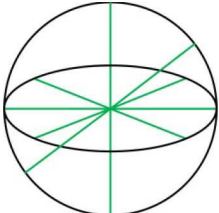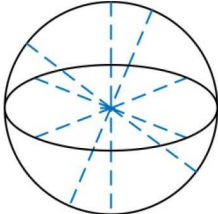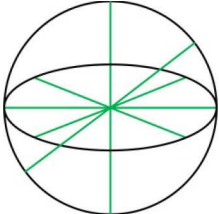
TABLE 1 | Line graph of the constraint spaces and freedom spaces.

Structure	Constraint space	Freedom space
UPS branches	No constraint	Three-dimensional rotation and three-dimensional translation
RRR branch	Three-dimensional constraint force	Three-dimensional rotation
Moving platform	Three-dimensional constraint force	Three-dimensional rotation

to the known Tricept mechanism, two unconstrained UPS branches provide six DOFs to the moving platform while three constraint force line vectors that through one point in space are acted in the platform wrench system via the properly constrained RRR branch, and thus retain three rotational ones of six DOFs. The line graph of the constraint spaces and freedom spaces of the branches and moving platform is presented in **Table 1**

based on Grassmann line geometry, in which the green solid lines, green solid double arrow lines, and blue dotted lines indicate the rotational DOF, translational DOF, and constraint force, respectively.

The K-G formula (Huang et al., 2013) is used to verify the number of DOFs obtained from the aforementioned analysis. The wearable parallel robot consists of a base, a moving platform, a

constraint branch and two kinematic branches:

$$F = 6(n - g - 1) + \sum_{i=1}^{g} f_i \tag{1}$$

where F is the DOFs of the robot, n indicates the number of links included in the frame, g represents the number of joints, and f_i is the DOFs permitted by joint i.

Since $n = 6$, $g = 7$, $\Sigma f_i = 15$ (the robot contains three spherical joints, two universal joints and two prismatic joints), the DOFs of the parallel mechanism can be obtained by equation 1 as $F = 3$.

Kinematic Analysis

The schematic diagram of the parallel robot is presented in **Figure 3**. Several reference frames, i.e., $O\text{-}x_o y_o z_o$, $O\text{-}x_p y_p z_p$, $B_i\text{-}x_{i1} y_{i1} z_{i1}$, $B_i\text{-}x_{i2} y_{i2} z_{i2}$, and $A_i\text{-}x_{i3} y_{i3} z_{i3}$, are established. Reference frames $O\text{-}x_o y_o z_o$ and $O\text{-}x_p y_p z_p$ (**Figure 3A**), which attached to rotation center O of the moving platform, are the base frame and the moving platform frame, respectively. These two reference frames are parallel in their initial configurations. As shown in **Figure 3B**, a local fixed reference frame $B_i\text{-}x_{i1} y_{i1} z_{i1}$ $(i = 1, 2)$ and a movable reference frame $B_i\text{-}x_{i2} y_{i2} z_{i2}$ $(i = 1, 2)$ are both assigned at the center B_i of joint U_i. Axes x_{i1}, y_{i2}, z_{i2} are collinear to the two revolute axes (r_{i1} and r_{i2}) of joint U_i and the lower link of the kinematic branch, respectively. In the set-up configuration, the axis y_{i1} coincides with the axis r_{i2} of joint U_i, and initial angles appear between the frames $B_i\text{-}x_{i1} y_{i1} z_{i1}$ and $B_i\text{-}x_{i2} y_{i2} z_{i2}$. Moreover, reference frame $A_i\text{-}x_{i3} y_{i3} z_{i3}$ $(i = 1, 2)$ in joint S_i location is assigned at the center A_i, and the directions of its three coordinate axes coincide with those of the frame $B_i\text{-}x_{i2} y_{i2} z_{i2}$. In this paper, due to the three rotational DOFs of the wearable parallel robot, the posture of the moving platform can be described by its orientation with respect to the base.

Considering the characteristics of the constraint branch, the $Z\text{-}X\text{-}Y$ typed Euler angles can be utilized to express the orientation of the moving platform. The angles rotating about axis z_p (AD/AB), axis x_p (PL/DO) and axis y_p (IN/EV) are denoted as γ, α, and β, respectively. The transformation from reference frame $O\text{-}x_p y_p z_p$ to $O\text{-}x_o y_o z_o$, denoted as matrix $\boldsymbol{R}_{op}$, is given as:

$$\boldsymbol{R}_{op} = \mathbf{R}(\gamma)\mathbf{R}(\alpha)\mathbf{R}(\beta) \tag{2}$$

Inverse Position Solution

The inverse kinematics problem posed by a parallel mechanism is easy to manage (Li et al., 2018), in which a desired posture of the moving platform is given, and the drive variables can be calculated to achieve this task.

The coordinates of point A_i $(i = 1, 2)$ can be computed simultaneously as follows:

$$^o\mathbf{A}_i = \mathbf{R}_{op}{}^P\mathbf{A}_i \tag{3}$$

$$^o\mathbf{A}_i = d_i \mathbf{z}_{i2} + {}^o\mathbf{B}_i = d_i \mathbf{R}_{o2}(0, 0, 1)^T + {}^o\mathbf{B}_i \tag{4}$$

where $^{\circ}\mathbf{A}_i$ and $^P\mathbf{A}_i$ are the position vector of point A_i with respect to reference frames $O\text{-}x_o y_o z_o$ and $O\text{-}x_p y_p z_p$, respectively. $^o\mathbf{B}_i$ $(i = 1, 2)$ denotes the position vector of point B_i expressed in reference frame $O\text{-}x_o y_o z_o$. d_i is the displacement of joint P_i, whereas $\mathbf{z}_{i2}$ is the direction vector of axis z_{i2}. The transformation matrix $\mathbf{R}_{o2}$ that transfers the coordinates from reference frame $O\text{-}x_o y_o z_o$ to $B_i\text{-}x_{i2} y_{i2} z_{i2}$ can be expressed as follows:

$$\mathbf{R}_{o2} = \mathbf{R}_{o1}\mathbf{R}_{12} \tag{5}$$

where $\mathbf{R}_{o1}$ and $\mathbf{R}_{12}$ represent the transformation matrices between reference frames $O\text{-}x_o y_o z_o$ and $B_i\text{-}x_{i1} y_{i1} z_{i1}$, $B_i\text{-}x_{i1} y_{i1} z_{i1}$, and $B_i\text{-}x_{i2} y_{i2} z_{i2}$, respectively, and are given as follows:

$$\mathbf{R}_{o1} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

$$\mathbf{R}_{12} = \begin{bmatrix} \cos(\theta_{i2}) & 0 & \sin(\theta_{i2}) \\ \sin(\theta_{i1})\sin(\theta_{i2}) & \cos(\theta_{i1}) & -\cos(\theta_{i2})\sin(\theta_{i1}) \\ -\cos(\theta_{i1})\sin(\theta_{i2}) & \sin(\theta_{i1}) & \cos(\theta_{i1})\cos(\theta_{i2}) \end{bmatrix} \tag{6}$$

where θ_{i1} and θ_{i2} denote the rotation angles of joint U_i around axes r_{i1} and r_{i2}, respectively.

By substituting the first half of Equation (4) into Equation (3), individual limb length d_i can be mathematically expressed as follows:

$$d_i = |\mathbf{R}_{op}{}^P\mathbf{A}_i - {}^o\mathbf{B}_i| \tag{7}$$

The rotation angle θ_3 around axis R_1 can be simply obtained as:

$$\theta_3 = \gamma \tag{8}$$

Equations (7) and (8) give the inverse position solution of the robot.

Substituting the second half of Equation (4) into Equation (3), and using Equations (5–7), the following equation can be derived:

$$\frac{\mathbf{R}_{o1}^{-1}(\mathbf{R}_{op}{}^P\mathbf{A}_i - {}^o\mathbf{B}_i)}{|\mathbf{R}_{op}{}^P\mathbf{A}_i - {}^o\mathbf{B}_i|} = \begin{bmatrix} \sin(\theta_{i2}) \\ -\cos(\theta_{i2})\sin(\theta_{i1}) \\ \cos(\theta_{i1})\cos(\theta_{i2}) \end{bmatrix} \tag{9}$$
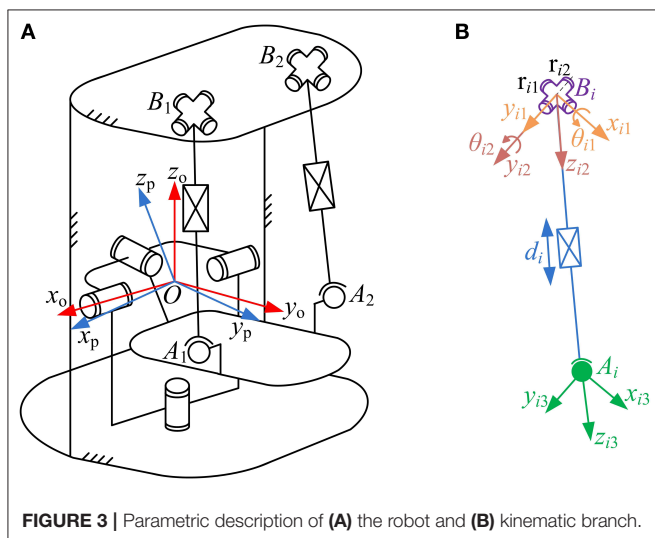
FIGURE 3 | Parametric description of **(A)** the robot and **(B)** kinematic branch.

Let k_{i1}, k_{i2}, k_{i3} indicate the three components of the left vector in equation 9, the following equations can be obtained as follows:

$$\theta_{i2} = \arcsin(k_{i1}), \theta_{i1} = \frac{1}{2}\arccos(\frac{k_{i2}^2 - k_{i3}^2}{1 - k_{i1}^2}) \qquad (10)$$

Velocity Jacobian Matrix

According to the schematic diagram of the robot presented in **Figure 3**, the velocity vector $\mathbf{V}_{A_i}$ of center A_i can be written as:

$$\mathbf{V}_{A_i} = \omega \times (\mathbf{R}_{0p}{}^p\mathbf{A}_i) \qquad (11)$$

where ω indicates the angular velocity of the moving platform.

Projecting the velocity vector $\mathbf{V}_{A_i}$ onto the reference frame A_i-$x_{i3}y_{i3}z_{i3}$ leads to:

$$\begin{bmatrix} \mathbf{V}_{x_{i3}} \\ \mathbf{V}_{y_{i3}} \\ \mathbf{V}_{z_{i3}} \end{bmatrix} = \begin{bmatrix} \mathbf{x}_{i3}^T \\ \mathbf{y}_{i3}^T \\ \mathbf{z}_{i3}^T \end{bmatrix} \mathbf{V}_{A_i} \qquad (12)$$

where $\mathbf{x}_{i3}^T$, $\mathbf{y}_{i3}^T$, and $\mathbf{z}_{i3}^T$ denote the direction vectors of axes x_{i3}, y_{i3}, and z_{i3}, respectively, and can be written as follows:

$$\mathbf{x}_{i3} = \mathbf{x}_{i2} = \mathbf{R}_{o2}(1, 0, 0)^T, \mathbf{y}_{i3} = \mathbf{y}_{i2} = \mathbf{R}_{o2}(0, 1, 0)^T,$$
$$\mathbf{z}_{i3} = \mathbf{z}_{i2} = \mathbf{R}_{o2}(0, 0, 1)^T \qquad (13)$$

where $\mathbf{x}_{i2}$, $\mathbf{y}_{i2}$, and $\mathbf{z}_{i2}$ represent the respective direction vectors of axes x_{i2}, y_{i2}, and z_{i2}.

The velocity of the linear actuator can be calculated based on the structural feature of the kinematic branch as:

$$\dot{d}_i = V_{z_{i3}} = \mathbf{z}_{i3}^T\mathbf{V}_{A_i} = \mathbf{z}_{i3}^T\mathbf{Q}\omega \qquad (14)$$

where $V_{z_{i3}}$ indicates the velocity component of $\mathbf{V}_{A_i}$ in the axis z_{i3} direction, $\dot{d}_i$ is the linear velocity of joint P_i. Moreover, coefficient matrix $\mathbf{Q}$ of $\mathbf{V}_{A_i}$ can be derived as follow:

$$\mathbf{Q} = \begin{bmatrix} 0 & \begin{pmatrix}0\\0\\1\end{pmatrix}^T (\mathbf{R}_{op}{}^p\mathbf{A}_i) & -\begin{pmatrix}0\\1\\0\end{pmatrix}^T (\mathbf{R}_{op}{}^p\mathbf{A}_i) \\ -\begin{pmatrix}0\\0\\1\end{pmatrix}^T (\mathbf{R}_{op}{}^p\mathbf{A}_i) & 0 & \begin{pmatrix}1\\0\\0\end{pmatrix}^T (\mathbf{R}_{op}{}^p\mathbf{A}_i) \\ \begin{pmatrix}0\\1\\0\end{pmatrix}^T (\mathbf{R}_{op}{}^p\mathbf{A}_i) & -\begin{pmatrix}1\\0\\0\end{pmatrix}^T (\mathbf{R}_{op}{}^p\mathbf{A}_i) & 0 \end{bmatrix}$$
$$\qquad (15)$$

The angular velocity $\dot{\theta}_3$ can be simply obtained as follows:

$$\dot{\theta}_3 = \dot{\gamma} \qquad (16)$$

Combining Equation (14) and Equation (16), the velocity mapping relationship between active joint space and task space can be expressed as follows:

$$\begin{bmatrix} \dot{d}_1 \\ \dot{d}_2 \\ \dot{\theta}_3 \end{bmatrix} = \begin{bmatrix} \mathbf{z}_{13}^T\mathbf{Q} \\ \mathbf{z}_{23}^T\mathbf{Q} \\ 0\ 0\ 1 \end{bmatrix} \omega = \mathbf{J}_o^{-1}\omega \qquad (17)$$

where $\mathbf{J}_o$ is the original velocity Jacobian matrix of the robot.

Notably, the input end contains two linear motions and one rotational motion, while the output end consists exclusively of rotational motions (i.e., the velocity Jacobian matrix is dimensionally inhomogeneous). Thus, a non-dimensional form (Angeles, 1992) of the homogeneous Jacobian matrix $\mathbf{J}_v$ is required to be introduced:

$$\mathbf{J}_v^{-1} = \begin{bmatrix} \mathbf{z}_{13}^T\mathbf{Q}^* \\ \mathbf{z}_{23}^T\mathbf{Q}^* \\ 0\ 0\ 1 \end{bmatrix} \qquad (18)$$

where:

$$\mathbf{Q}^* = \begin{bmatrix} 0 & \begin{pmatrix}0\\0\\1\end{pmatrix}^T (\mathbf{R}_{op}{}^p\mathbf{A}_i^*) & -\begin{pmatrix}0\\1\\0\end{pmatrix}^T (\mathbf{R}_{op}{}^p\mathbf{A}_i^*) \\ -\begin{pmatrix}0\\0\\1\end{pmatrix}^T (\mathbf{R}_{op}{}^p\mathbf{A}_i^*) & 0 & \begin{pmatrix}1\\0\\0\end{pmatrix}^T (\mathbf{R}_{op}{}^p\mathbf{A}_i^*) \\ \begin{pmatrix}0\\1\\0\end{pmatrix}^T (\mathbf{R}_{op}{}^p\mathbf{A}_i^*) & -\begin{pmatrix}1\\0\\0\end{pmatrix}^T (\mathbf{R}_{op}{}^p\mathbf{A}_i^*) & 0 \end{bmatrix},$$
$${}^p\mathbf{A}_i^* = k^* \frac{{}^p\mathbf{A}_i}{r_{sp}} \qquad (19)$$

where $p_i^{\mathbf{A}} *$ is the position vector of point A_i with respect to the reference frame O-$x_py_pz_p$, expressed in non-dimensional form. A scalar r_{sp} indicates the distribution radius of the spherical joint with respect to the moving platform and is utilized to homogenize the original velocity Jacobian matrix (Zanganeh and Angeles, 1997). k^* represents the scaling factor between the linear motion and rotational motion (generally, $k^* = 1$).

Performance Indices
Reachable Workspace Index

The reachable workspace of an ankle rehabilitation robot must contain the ROM of the HAC summarized (Siegler et al., 1988) in **Table 2**. Meanwhile, to ensure that the safety issue will not occur in the process of treatment, the maximum allowable workspace (MAW) of the ankle rehabilitation robot should be constrained in a certain range (**Table 2**). By using the numerical searching method based on derived inverse position solution while considering the stroke constraint of the linear actuator, the feasible points in the reachable workspace of the ankle rehabilitation robot with certain dimension parameters can be obtained, and then the set of the reachable points forms

Type of motion	ROM/(°)	MAW of the robot /(°)
Plantarflexion	37.6–45.8	45.0
Dorsiflexion	20.3–29.8	30.0
Inversion	14.5–22.0	22.0
Eversion	10.0–17.0	22.0
Adduction	22.0–36.0	36.0
Abduction	15.4–25.9	36.0

the overall workspace. To evaluate the workspace, a reachable workspace index I_{RW} can be written as follows:

$$I_{RW} = \frac{v_{RW}}{v_{MAW}}, \tag{20}$$

$$v_{RW} = \int_w dw, \tag{21}$$

$$v_{MAW} = \Delta\alpha\Delta\beta\Delta\gamma, \tag{22}$$

where v_{RW} and v_{MAW} are the volume of the reachable workspace and the MAW, respectively, $\Delta\alpha$, $\Delta\beta$, and $\Delta\gamma$ denote the ranges between the minimum and maximum α, β, and γ, respectively, which can be given as follows:

$$\Delta\alpha = \alpha_{max} - \alpha_{min}, \tag{23}$$

$$\Delta\beta = \beta_{max} - \beta_{min}, \tag{24}$$

$$\Delta\gamma = \gamma_{max} - \gamma_{min}, \tag{25}$$

where α_{max}, α_{min}, β_{max}, β_{min}, γ_{max}, and γ_{min} can be obtained according to **Table 2**. The I_{RW} can reach values from 0 to 1. The value is equal to 1 (or 0) mean that the robot possesses the largest (or smallest) workspace.

Motion Isotropy Index

The inverse value of the condition number of robot's velocity Jacobian matrix, ranges between 0 and 1 (denote singular and isotropic configuration, respectively), is an important local performance index to evaluate the motion isotropy in one posture or over its full workspace of a parallel robot (Wu et al., 2013; Enferadi and Nikrooz, 2017). Its physical meaning can be expressed as a velocity ellipsoid and define as Equation (26). For a rehabilitation device, as many areas as possible in the reachable workspace are desired to possess relatively uniform motion isotropy. That is, condition number's inverse value of most of the feasible points should be closer to 1. To measure the global behavior of the condition number of the robot, a motion isotropy index I_{MI} can be presented via computing the average of the inverse value of the condition number within the reachable workspace, and is written as follows:

$$\dot{\mathbf{S}}^T\dot{\mathbf{S}} = \omega^T\left(\mathbf{J}_v\mathbf{J}_v^T\right)^{-1}\omega \leq 1, \tag{26}$$

$$I_{MI} = \frac{\int_w \eta_J dw}{v_{RW}}, \tag{27}$$

$$\eta_J = l_{vsp}/l_{vlp} \tag{28}$$

where $\dot{\mathbf{S}} = \left(\dot{d}_1, \dot{d}_2, \dot{\theta}_3\right)^T$; w denotes the reachable workspace of the ankle rehabilitation robot; η_J is a local index indicating the inverse value of the condition number of robot's velocity Jacobian matrix in a given posture within the reachable workspace; l_{vlp} and l_{vsp} are the lengths of the long and short principal axes of the velocity ellipsoid, respectively. The value range of the I_{MI} is between 0 and 1, and the value of which is desired to be larger.

Force Tansfer Index

As a human-robot system for ankle rehabilitation, the force is required to be transferred from robot's active joint space to patient's ankle space as sufficient torque which is an important condition for an ankle rehabilitation robot to achieve passive/active treatment. A force unit sphere $f^Tf \leq 1$ is set up in active joint space. Subsequently, this sphere can be transferred into task space as a force ellipsoid via the force mapping relationship, and can be defined as follows:

$$f = \mathbf{J}_f\tau = \mathbf{J}_v^T\tau, \tag{29}$$

$$f^Tf = \tau^T(\mathbf{J}_f^T\mathbf{J}_f)\tau = \tau^T(\mathbf{J}_v\mathbf{J}_v^T)\tau \leq 1, \tag{30}$$

where $f = (f_1, f_2, \tau_3)^T$ and $\tau = (\tau_\alpha, \tau_\beta, \tau_\gamma)^T$; f_1 and f_2 are the driving forces of joints P_1 and P_2, and τ_3 is the driving torque of joint R_1; τ_α, τ_β, and τ_γ indicate the torques applied on the axes of PL/DO, IN/EV, and AB/AD, respectively; $\mathbf{J}_f$ denotes the force Jacobian matrix, and is the transpose of Jacobian matrix $\mathbf{J}_v$.

The robot possesses a better (or worse) force transfer performance along a particular operation direction when the length of the force ellipsoid's radius along the directional vector is longer (or shorter). Moreover, the long (or short) principal directions of the force ellipsoid means the greatest (or least) force transfer performance. Thus, the length l_{fsp} of the short principal axis of the force ellipsoid can be regarded as a local evaluation index of force transfer performance, and the corresponding global force transfer index I_{FT} is given as follows:

$$I_{FT} = \frac{\int_w l_{fsp} dw}{v_{RW}}. \tag{31}$$

Maximum Torque Index

To evaluate the force capability of the ankle rehabilitation robot while considering the real physical capability of robot's actuators, a set $\mathbf{T}_\tau$ (or convex polyhedrons) of allowable forces and torques of the actuators should be defined based on the force Jacobian matrix in the task space. The radius r_{is} of the inscribed sphere contained in the set indicates the largest real torque that can be realized by the ankle rehabilitation robot along all directions in the ankle space, i.e., this radius reflects the maximum torque in a given posture. According to the aforementioned analysis and the force mapping relationship, the set $\mathbf{T}_\tau$, and a global maximum torque index I_{MT} of the ankle rehabilitation robot can be written as follows:

$$\mathbf{T}_\tau = \left\{ \tau \mid \tau = (\mathbf{J}_f)^{-1}f \ f \in T_f \right\}, \tag{32}$$

$$T_f = \{\boldsymbol{f} \mid |f_i| \leq f_{i\max} \ i = 1, 2; \ |\tau_3| \leq \tau_{3\max}\} \ , \quad (33)$$

$$I_{MT} = \frac{\int_w l_{ris} dw}{v_{RW}} \ , \quad (34)$$

where T_τ is the generalized set of torques in the ankle space; T_f is the allowable forces and torques of the actuator; l_{ris} is a local index denoting the length of the r_{is}, and is actually a local performance index. Without loss of generality, the force and torque limits of the driven system are assumed to be $f_{1\max} = f_{2\max} = \tau_{3\max} = 1$. Thus, the I_{MT} ranges from 0 to 1, a larger (or smaller) value of I_{MT} indicates a better (or worse) force capability of the ankle rehabilitation robot.

RESULTS

Mechanical Design

Based on the proposed kinematic configuration (i.e., 2-UPS/RRR parallel mechanism) and wearable design concept, the mechanical design of the wearable parallel robot (**Figure 4**) is detailed. Two sub-systems, i.e., a basic machine-drive system and a multi-model position/force data collection system, constitute the whole physical system of the robot.

The machine-drive system exhibits a parallel main structure, in which two kinematic branches and one constraint branch both connect the base to the moving platform. The base consists of a base plate and a supporting column. An adjustable lower-limb binding mechanism is established between the supporting column to maintain the stability of the lower-limb during rehabilitation and accommodate patients with different body sizes (**Figure 5**), double linear guide rails with double-slider, single-connection platform and locking function are employed to fix the thigh and adjust the up/down position of the patient, while the calf is fixed by a special leg holder and its forward/backward position can be fine-tuned by single linear guide rail with single-slider and locking function. Subsequently, the base and two identical kinematic branches are connected via the joint U_1 and U_2. Two linear actuators (CAHB-10, SKF, Sweden) are employed as joint P_1 and P_2 to adjust the lengths of the kinematic branches from 413 to 713 mm. As illustrated in **Figure 6**, joint S_1 (S_2) that located below the joint U_1 (U_2), and P_1 (P_2) is equivalent to a universal joint and a revolute joint with three axes intersecting at the same point, this combination design can reduce the cost and realize free-interference on workspace. Moreover, by loosening the screw bolt connecting the lower end of the linear actuator and joint S_1 (S_2), the kinematic branches can be separated from the constraint branch (i.e., the 2-UPS/RRR parallel robot is translated into an RRR serial robot). As shown in **Figure 7**, three lockable binding bands secure the patient's foot to the upper part of the moving platform (i.e., the upper platform) without large misalignment during combined motion. In addition to joint S_1 and S_2, the lower platform is also connected to joint R_3 via an "L" shaped frame. As the active vertical-revolute joint
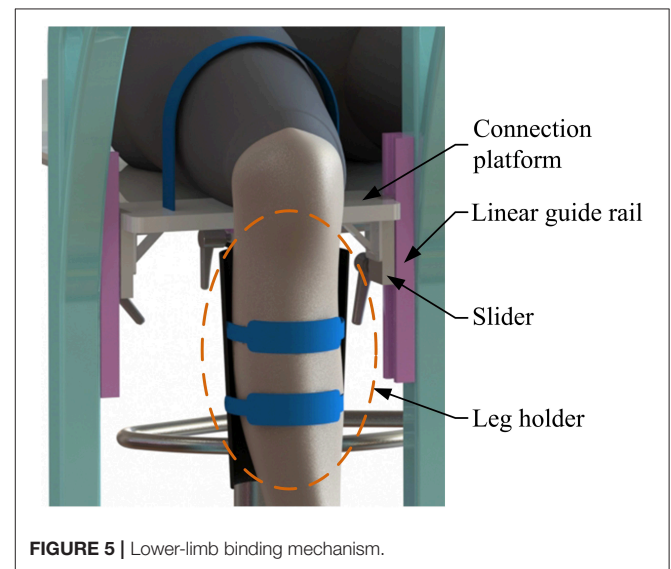
FIGURE 5 | Lower-limb binding mechanism.

FIGURE 4 | Wearable parallel robot.
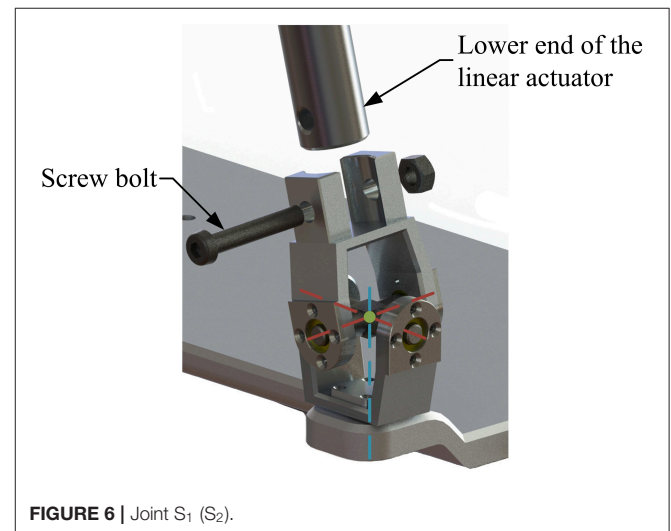
FIGURE 6 | Joint S_1 (S_2).

of the constraint branch, joint R_1 is driven by a servo motor (**Figure 8**), i.e., the combination of a frameless motor (KBM, Kollmorgen, America) and an incremental encoder (HKT30-301, REP, China), and is transmitted via a harmonic reducer (CSG-17-100, HarmonicDrive, Japan). Notably, the distance between the moving platform and the base plate determines the height of the patient's seat, and a higher seat may produce fear emotions that lead to negative treatment results on patient. Thus, to reduce the height of the moving platform, the axis R_1 and its driving unit (i.e., the servo motor and the incremental encoder) are arranged in parallel and are connected to each other by a synchronous belt with a 1:1 reduction ratio. Additionally, as shown in **Figure 8**, screw bolts are installed as mechanical limits on joint R_1, and two suspended revolute joints (i.e., joints R_2 and R_3) for safety. As mentioned above, the limits for the rotation angles are set according to the MAW in **Table 2**, and the maximum allowable angles of the PL/DO ($\alpha_{max}/\alpha_{min}$), IN/EV ($\beta_{max}/\beta_{min}$), and AD/AB ($\gamma_{max}/\gamma_{min}$) are set at 45°/30°, 22°/22°, and 36°/36° to ensure that the robot is suitable for both the left and right foot.
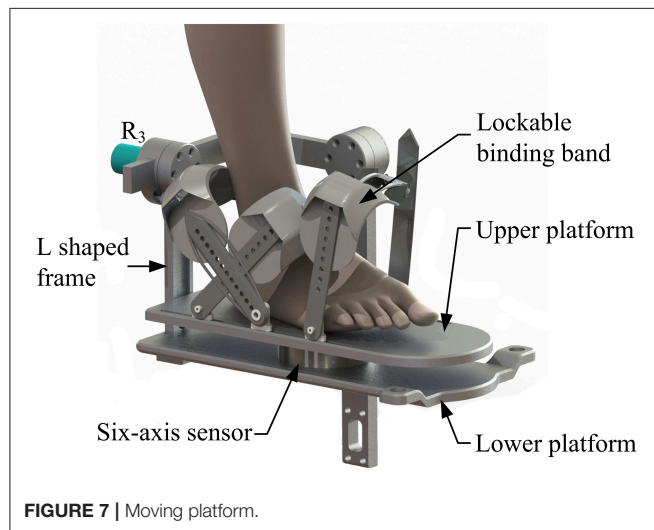
FIGURE 7 | Moving platform.

A multi-model position/force data collection system is installed in the robot to realize various rehabilitation strategies, including ROM treatment based on position control, strength treatment based on impedance control and proprioceptive treatment base on intention recognition. Specifically, as illustrated in **Figure 8**, three absolute rotary encoders (HAN28U5, China) are arranged to measure the rotation angles of joints R_1, R_2, and R_3 (i.e., real-time position information of the moving platform); two of them are connected in series with joints R_2 and R_3, as shown in the detailed view (**Figure 9**), joints R_2 and R_3 possess similar structure: the base, "C" shaped frame#1, "C" shaped frame#2, and "L" shaped frame are arranged in sequence, the latter rotates with respect to the former, and the "L" shaped frame drives the moving platform together with the "V" shaped part of the lower platform; another one is placed on the base plate and arranged in parallel with joint R_1. Additionally, the force/torque information of the rehabilitation process, i.e., the interaction force and torque between the foot and moving platform, can be collected by a six-axis sensor installed between the upper and lower platform (**Figure 7**). In general, the encoders and sensor form a complete information collection system, which can produce real-time feedback in the process of treatment and lay a foundation for various control schemes and rehabilitation strategies.

Performance Evaluation

The geometrical parameters of the wearable parallel robot are presented in **Table 3**, where the absolute values of the coordinates of joint U_i are expressed with respect to reference frame O-$x_o y_o z_o$, the absolute values of the coordinates of joint S_i are expressed with respect to reference frame O-$x_p y_p z_p$.

The reachable workspace (the set of the solid points) of the wearable parallel robot is calculated in **Figure 10A**. For this robot, the constraint condition is the stroke constraint of the linear actuators and the arrangement of the mechanical limits. According to the calculation results, the appearance of the reachable workspace is a cube, suggesting that the robot can reach any posture in MAW, i.e., the volume of the reachable workspace
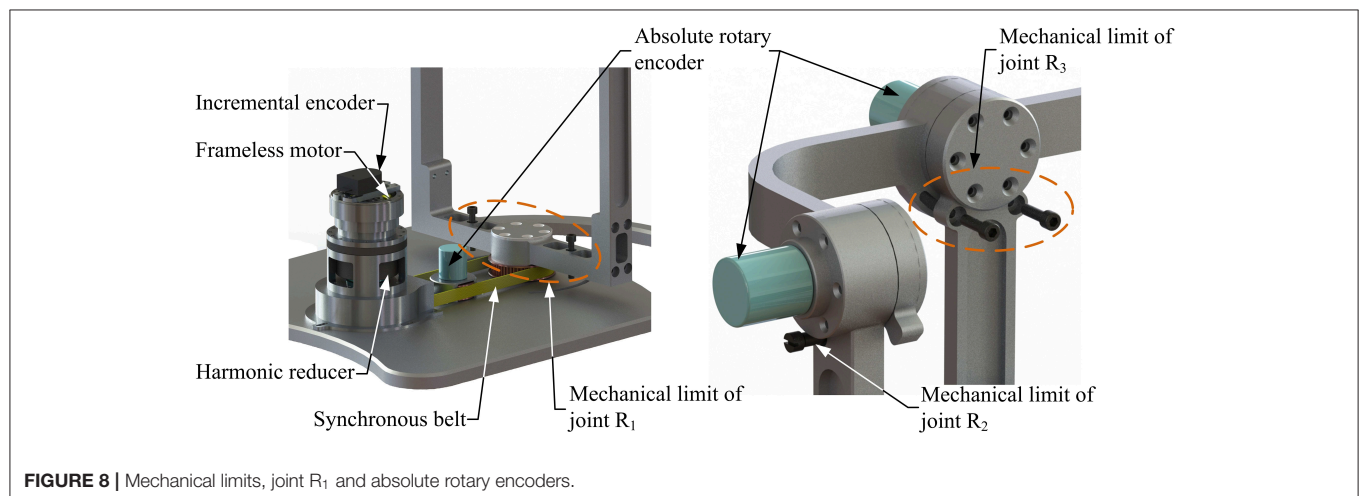
FIGURE 8 | Mechanical limits, joint R_1 and absolute rotary encoders.
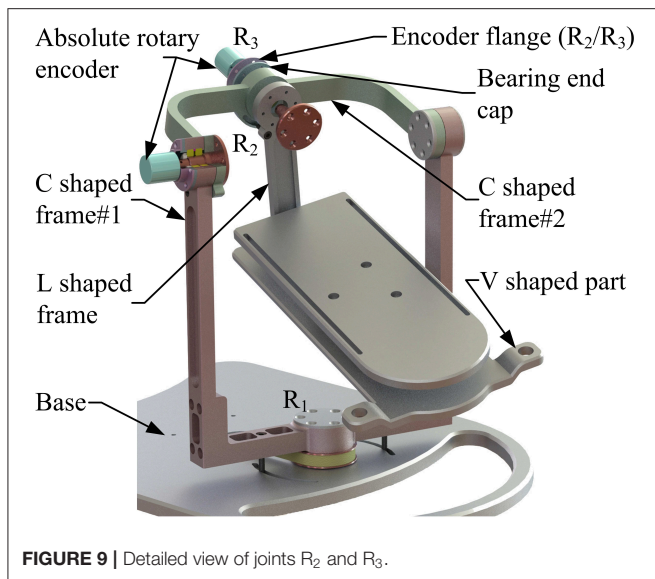
FIGURE 9 | Detailed view of joints R_2 and R_3.

TABLE 3 | The geometrical dimensions of the robot.

Robotic dimensions	Absolute values of coordinates		
	X	Y	Z
Center B_i of joint U_i	120 mm	70 mm	520 mm
Center A_i of joint S_i	90 mm	215 mm	107 mm

is equal to that of the MAW, and thus $I_{RW} = 1$. Additionally, when the mechanical limits are removed, the workspace of the robot is represented by a high transparency shadow (**Figure 10A**) which covers the set of the solid points, indicating that the mechanical limits can effectively restrict the workspace to a safe range.

The η_J, l_{spa}, and l_{ris} are evaluated within the calculated reachable workspace. Subsequently, trends of their corresponding values are illustrated in **Figures 10B–D**, respectively, with values represented by the color map shown in the color bar and trends represented by the color variation in the color map. The distribution volume proportions of the corresponding values in the reachable workspace are shown in **Figures 11A–C**, while the proportion that indices η_J, l_{fsp}, and l_{ris} superior to indices I_{MI}, I_{FT}, and I_{MT} is shown in **Figure 11D**.

As shown in **Figure 10B**, the values of η_J change smoothly with no mutation within overall reachable workspace, indicating that the robot has no singularity configuration. Moreover, the robot exhibits better motion isotropy performance in the central part of the reachable workspace, since the values of the η_J are relatively small in the boundary area and increases gradually toward the central section. As shown in **Figure 11A**, the value of η_J is mainly distributed between 0.3 and 0.6, and the minimum value, varies from 0.24 to 0.3, accounted for 6.5552% in the reachable workspace. Thus, the robot is sufficiently kinematically isotropic for ankle rehabilitation.

Analogous to η_J, both l_{fsp} and l_{ris} possess better performance in the central part (**Figures 10C,D**). As illustrated in **Figures 11B,C**, the values of l_{fsp} and l_{ris} are mainly distributed between 0.5 and 0.69, 0.6, and 0.91, respectively. Moreover, the proportions of the worst-performing postures of l_{fsp} and l_{ris} are 3.5050 and 10.7826%. Thus, the robot possesses high force transfer performance and large maximum torque performance, especially in the central part.

The values of I_{MI}, I_{FT}, and I_{MT} are calculated as $I_{MI} = 0.5573$, $I_{FT} = 0.5565$, and $I_{MT} = 0.6744$, demonstrating sufficient global performances. As shown in **Figure 11D**, the proportion of η_J, l_{fsp}, and l_{ris} exceeded I_{MI}, I_{FT}, and I_{MT} in magnitude are 41.2040, 56.5151, and 54.1271%, mainly located in the central section (i.e., main treatment area), suggesting that most postures within the reachable workspace are well-performed enough (although the proportion of η_J superior to indices I_{MI} is failure to reach 50%, most postures that do not meet the condition exceed 0.5 in magnitude, as shown in **Figure 10B**).

To verify the calculation correctness and obtain a detailed view, velocity ellipsoids, force ellipsoids, sets T_τ and their inscribed spheres in two stochastic configurations ([$\alpha = -30°$, $\beta = 20°$, $\gamma = 5°$], [$\alpha = 10°$, $\beta = -10°$, $\gamma = -2°$]) of the reachable workspace are provided in **Figure 12**. **Figures 12A–D** report that due to the duality between the velocity ellipsoid and the force ellipsoid (Chiu, 1988), the principal axial directions of the two ellipsoids are coincident and the lengths, i.e. l_{vsp} and l_{flp} (the length of the long principal axis of the force ellipsoid), l_{vlp} and l_{fsp}, are reciprocal. Index η_J are calculated as 0.4081 and 0.3866 in these two configurations, while index l_{spa} are obtained as 0.4477 and 0.6153, respectively. **Figures 12E,F** indicate that the cube in active joint space maps to an irregular polyhedron in task space with an inscribed sphere tangent to the colored surfaces, and index l_{ris} are, respectively calculated as 0.4790 and 0.7797.

DISCUSSIONS

We introduced a novel robot with the features of a wearable design concept and parallel platform-based form. The salient advantages of this robot are its simple configuration and safety guarantee. Moreover, we evaluated and analyzed the performance of this robot within the overall reachable workspace. Results showed that the proposed robot possesses sufficient motion isotropy, high force transfer performance and large maximum torque performance.

In comparison with the ankle rehabilitation robot applying redundant actuation, by adopting a simple kinematic configuration (i.e., 2-UPS/RRR parallel mechanism) as the main mechanical structure, actuator non-redundancy and easy operation can be realized on this wearable parallel robot, and then the cost of the robot manufacturing, the difficulty of the control system development, and the burden of the therapists can also be effectively reduced. Additionally, according to the performance evaluation and analysis, the proposed robot satisfies the conditions of performing ankle rehabilitation treatment for patients.
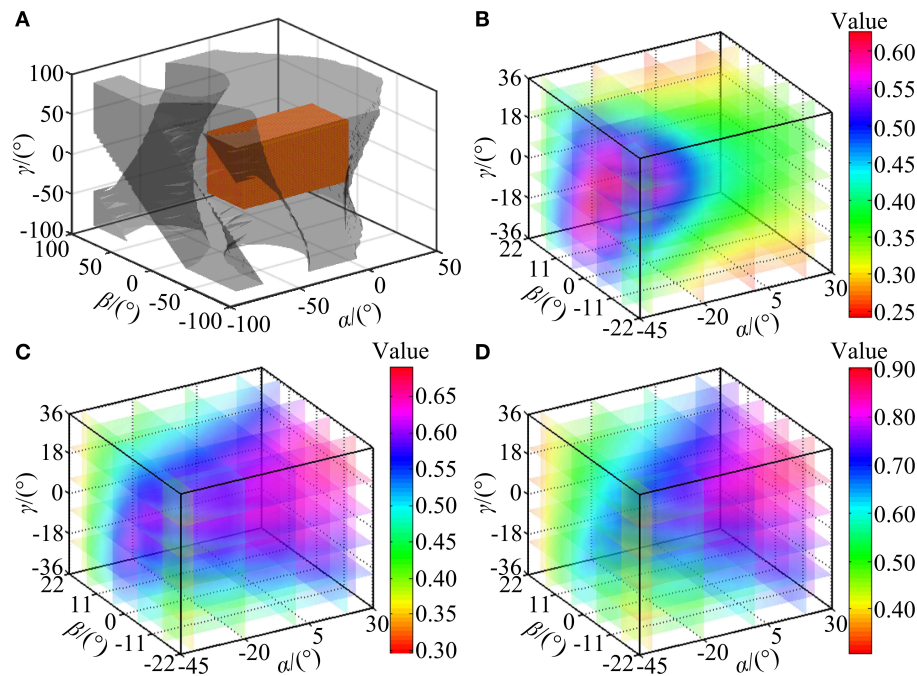
FIGURE 10 | Performance indices calculation. **(A)** The reachable workspace of the robot. **(B)** Trend in value of η_J. **(C)** Trend in value of l_{fsp}. **(D)** Trend in value of l_{ris}.
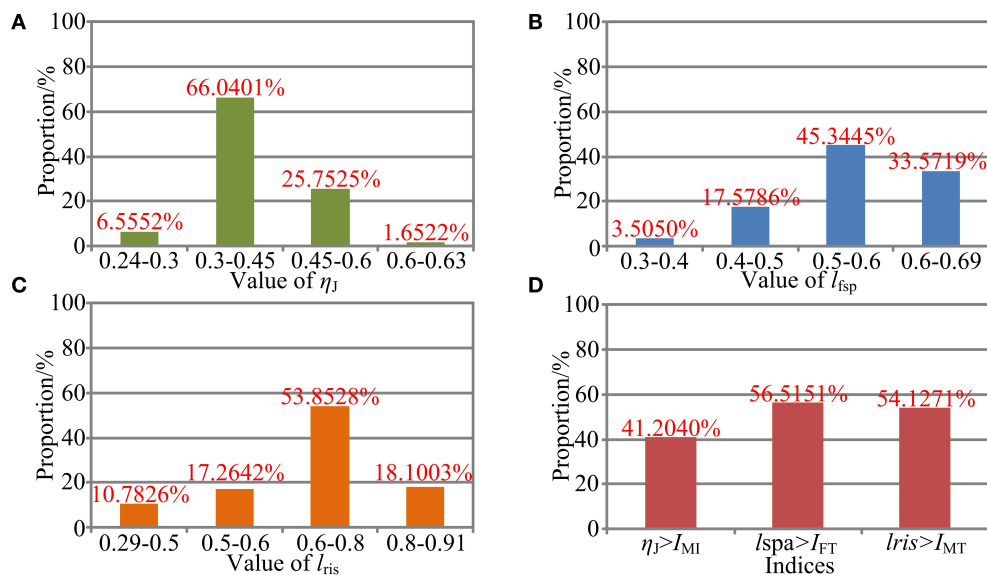
FIGURE 11 | Performance evaluation. **(A)** Performance distribution of η_J. **(B)** Performance distribution of l_{fsp}. **(C)** Performance distribution of l_{ris}. **(D)** Performance comparison between the local indices and the global indices.

This wearable parallel robot is designed with several safety precautions to protect the patients from secondary injury in the process of treatment. Specifically, this robot possesses the lowest moving platform under the premise of meeting the ROM requirement, and then correspondingly reduces the height of the patient's seat. An excessively high seat increases the risk of injury and easily results in patients' contravene mood. Moreover,

the arrangement of the lower-limb binding mechanism and three lockable binding bands, respectively fix the thigh/calf and the foot with the base and the moving platform. With this wearable design concept, the foot moves in the form of platform-base with respect to the lower-limb, and thus avoiding the coupled foot-shank motion. Additionally, the supporting column is narrow in width to allow the patient to place the
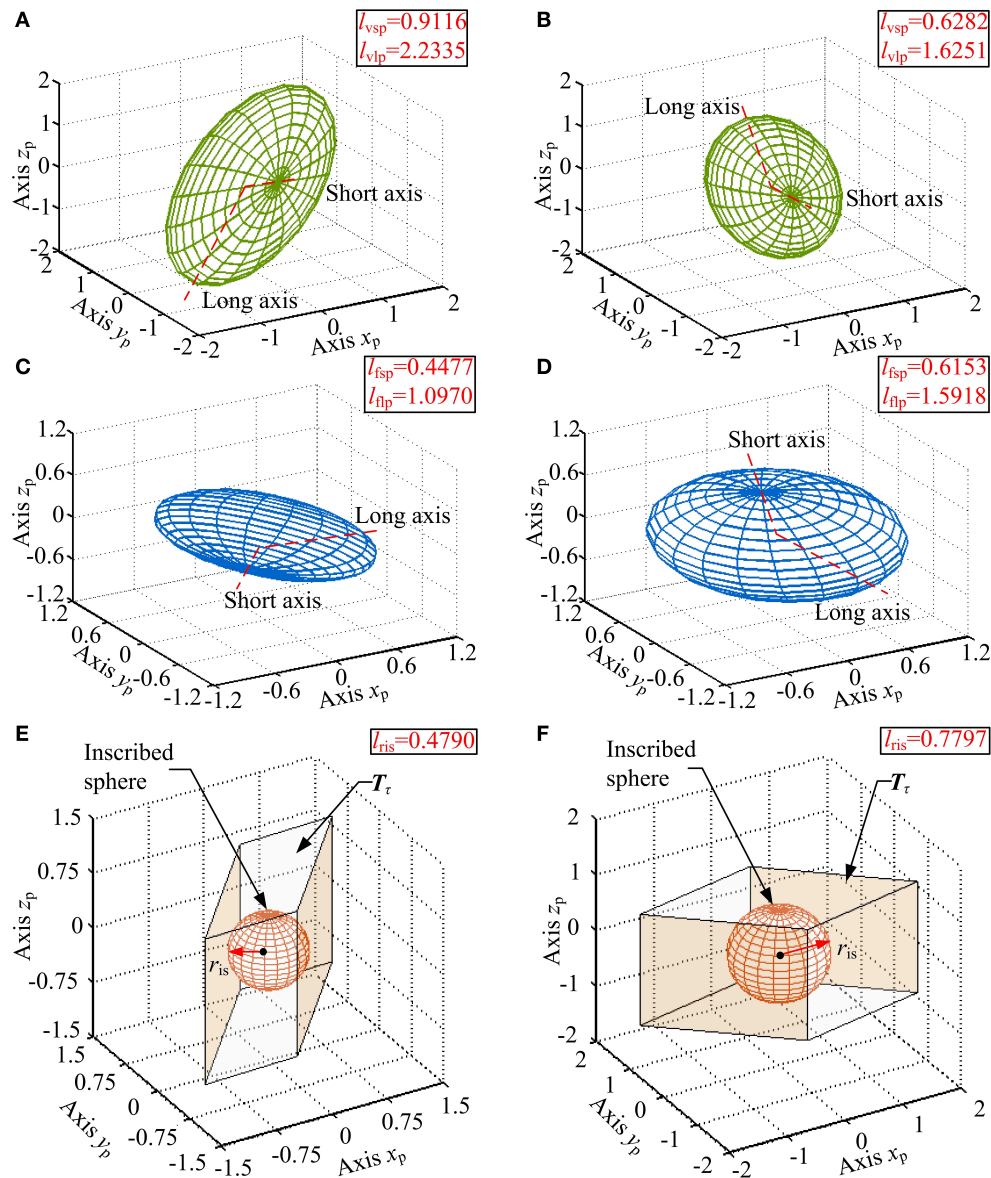
FIGURE 12 | (A,B) Velocity ellipsoids, **(C,D)** force ellipsoids, **(E,F)** sets $\boldsymbol{T}_\tau$ and their inscribed spheres for $[\alpha = -30°, \beta = 20°, \gamma = 5°]$, $[\alpha = 10°, \beta = -10°, \gamma = -2°]$ configurations.

non-rehabilitation leg conveniently. Demountable mechanical limits are implemented to constrain the workspace of the robot in a safe range.

In terms of information collection, the proposed robot is equipped with multi-model position/force data collection system containing a six-axis sensor and three absolute rotary encoders. Based on the collected kinematic and dynamic information, passive treatment mode and active treatment mode can be achieved. By separating the spherical joint from the upper end of the linear actuators, the 2UPS/RRR parallel robot can be translated into an RRR serial robot. Therapists can bind the patient's foot on the moving platform in advance and plan the treatment trajectory by manually moving the RRR

serial robot according to the joint characteristics and the severity of impairment among different patients. The trajectory can be recorded by the three absolute rotary encoders, and the robot can provide repetitive treatment on the basis of this trajectory.

It should be highlight that the indices are not applied to design the geometrical parameters of the mechanism in this study. A low height of the moving platform, a narrow width of the overall mechanism and a sufficient volume of the reachable workspace are more considered at current stage. Moreover, by applying the transmission angular theory and screw theory to evaluate the motion/force transmissibility (i.e., essential function) of the parallel mechanism, a more convincing

kinematic performance evaluation system (Liu et al., 2008; Wu et al., 2010, 2011; Xie et al., 2010) was proposed compared with the utilization of the Jacobian matrix in evaluation of parallel mechanism. Based on the defined motion/force transmission indices, dimensional synthesis (optimal design) was also carried out using performance charts. In future studies, dimension optimal design according to the aforementioned method will be carried out before prototype construction. Additionally, based on the multi-model position/force data collection system, future work could go in the direction of the development of the control schemes to achieve various rehabilitation protocols.

CONCLUSION

This paper presented a novel wearable parallel robot for ankle rehabilitation in which the intended simple configuration is determined according to the HAC anatomy and safety consideration. Based on the performance evaluation, the proposed robot possesses relatively uniform motion isotropy, high force transfer performance and large maximum torque performance within a large reachable workspace. Equipped with a multi-model position/force data collection system, both passive and active treatment mode can be achieved. And the robot has the potential to be used for the rehabilitation treatment of ankle disabilities.

DATA AVAILABILITY STATEMENT

AUTHOR CONTRIBUTIONS

SZ and JL conceived and designed this study. SZ, XZ, WF, and YK performed the mechanical design and performance evaluation. SZ wrote the paper. MD reviewed and edited the manuscript. JL also made a contribution to the edition of the manuscript. All authors had read and approved the manuscript.

FUNDING

REFERENCES

Angeles, J. (1992). The design of isotropic manipulator architectures in the presence of redundancies. *Int. J. Robot. Res.* 11, 196–201. doi: 10.1177/027836499201100303

Blaya, J. A., and Herr, H. (2004). Adaptive control of a variable-impedance ankle-foot orthosis to assist drop-foot gait. *IEEE Trans. Neural Syst. Rehabil. Eng.* 12, 24–31. doi: 10.1109/TNSRE.2003.823266

Boehler, A. W., Hollander, K. W., Sugar, T. G., and Shin, D. (2008). "Design, implementation and test results of a robust control method for a powered ankle foot orthosis (AFO)," in *2008 IEEE International Conference on Robotics and Automation (ICRA)* (Pasadena, CA), 2025–2030. doi: 10.1109/ROBOT.2008.4543504

Chiu, S. L. (1988). Task compatibility of manipulator postures. *Int. J. Robot. Res.* 7, 13–21. doi: 10.1177/027836498800700502

Dai, J. S., and Zhao, T. (2004). Sprained ankle physiotherapy based mechanism synthesis and stiffness analysis of a robotic rehabilitation device. *Auton. Robots* 16, 207–218. doi: 10.1023/B:AURO.0000016866.80026.d7

Dettwyler, M., Stacoff, A., Quervain, I. A. K., and Stussi, E. (2004). Modeling of the ankle joint complex. Reflections with regards to ankle prostheses. *Foot Ankle Surg.* 10, 109–119. doi: 10.1016/j.fas.2004.06.003

Dijk, W. V., Meijneke, C., and Kooij, H. V. D. (2017). Evaluation of the Achilles ankle exoskeleton. *IEEE Trans. Neural Syst. Rehabil. Eng.* 25, 151–160. doi: 10.1109/TNSRE.2016.2527780

Dul, J., and Johnson, G. E. (1985). A kinematic model of the human ankle. *J. Biomed. Eng.* 7, 137–143. doi: 10.1016/0141-5425(85)90043-3

Enferadi, J., and Nikrooz, R. (2017). The performance indices optimization of a symmetrical fully spherical parallel mechanism for dimensional synthesis. *J. Intell. Robot. Syst.* 90, 305–321. doi: 10.1007/s10846-017-0675-6

Erdogan, A., Celebi, B., Satici, A. C., and Patoglu, V. (2017). AssistON-Ankle: a reconfigurable ankle exoskeleton with series-elastic actuation. *Auton. Robot.* 41, 743–758. doi: 10.1007/s10514-016-9551-7

Fan, Y., and Yin, Y. (2009). "Mechanism design and motion control of a parallel ankle joint for rehabilitation robotic exoskeleton," in *Proceedings of the 2009 IEEE International Conference on Robotics and Biomimetics* (Guilin), 2527–2532. doi: 10.1109/ROBIO.2009.5420488

Ferris, D. P., Gordon, K. E., Sawicki, G. S., and Peethambaran, A. (2006). An improved powered ankle-foot orthosis using proportional myoelectric control. *Gait Posture* 23, 425–428. doi: 10.1016/j.gaitpost.2005.05.004

Feuerbach, J. W., Grabiner, M. D., Koh, T. J., and Weiker, G. G. (1994). Effect of an ankle orthosis and ankle ligament anesthesia on ankle joint proprioception. *Am. J. Sport. Med.* 22, 223–229. doi: 10.1177/036354659402200212

Girone, M., Burdea, G., and Bouzit, M. (1999). "The rutgers ankle orthopedic rehabilitation interface," in *ASME Haptics Symposium*. Vol 67 (Nashville), 305–312.

Gordon, K. E., Sawicki, G. S., and Ferris, D. P. (2006). Mechanical performance of artificial pneumatic muscles to power an ankle-foot orthosis. *J. Biomech.* 39, 1832–1841. doi: 10.1016/j.jbiomech.2005.05.018

Hollander, K. W., Ilg, R., Sugar, T. G., and Herring, D. (2006). An efficient robotic tendon for gait assistance. *J. Biomech. Eng.* 128, 788–791. doi: 10.1115/1.2264391

Huang, Z., Li, Q., and Ding, H. (2013). *Theory of Parallel Mechanisms*. (Netherlands: Springer). doi: 10.1007/978-94-007-4201-7

Hussain, S., Jamwal, P. K., and Ghayesh, M. H. (2017). State-of-the-art robotic devices for ankle rehabilitation: mechanism and control review. *Proc. Inst. Mech. Eng. Part H* 231, 1224–1234. doi: 10.1177/0954411917737584

Isman, R. E., and Inman, V. T. (1969). Anthropometric studies of the human foot and ankle. *Bull. Pros. Res.* 11, 97–129.

Jamwal, P., Xie, S. Q., Hussain, S., and Parsons, J. (2014). An adaptive wearable parallel robot for the treatment of ankle injuries. *IEEE/ASME Trans. Mech.* 19, 64–75. doi: 10.1109/TMECH.2012.2219065

Khalid, Y., Gouwanda, D., and Parasuraman, S. (2015). A review on the mechanical design elements of ankle rehabilitation robot. *J. Eng. Med.* 229, 452–463. doi: 10.1177/0954411915585597

Kim, J., Hwang, S., Sohn, R., Lee, Y., and Kim, Y. (2011). Development of an active ankle foot orthosis to prevent foot drop and toe drag in hemiplegic patients: a preliminary study. *Appl. Bionics Biomech.* 8, 377–384. doi: 10.1155/2011/530375

Kinnaird, C. R., and Ferris, D. P. (2009). Medial gastrocnemius myoelectric control of a robotic ankle exoskeleton. *IEEE Trans. Neural. Syst. Rehabil. Eng.* 17, 31–37. doi: 10.1109/TNSRE.2008.2008285

Li, J., Li, S., Zhang, L., Tao, C., and Ji, R. (2018). Position solution and kinematic interference analysis of a novel parallel hip-assistive mechanism. *Mech. Mach. Theory* 120, 265–287. doi: 10.1016/j.mechmachtheory.2017.10.002

Liu, G., Gao, J., Yue, H., Zhang, X., and Lu, G. (2006). "Design and kinematics simulation of parallel robots for ankle rehabilitation," in *Proceedings of the 2006 IEEE International Conference on Mechatronics and Automation* (Beijing), 253–258. doi: 10.1109/ICMA.2006.257780

Liu, X., Wu, C., and Wang, J. (2008). "A new index for the performance evaluation of parallel manipulators: a study on planar parallel manipulators," in *Proceedings of 7th World Congress on Intelligent Control and Automation* (Chongqing), 353–357. doi: 10.1109/WCICA.2008.4592950

Lopez, R., Salazar, S., and Lozano, R. (2013). "Model and control of the elltio with two degrees of freedom," in *International Conference on System Theory, Control and Computing* (Sinaia), 305–310. doi: 10.1109/ICSTCC.2013.6688977

Meijneke, C., Dijk, W. V., and Kooij, H. V. D. (2014). "Achilles: an autonomous lightweight ankle exoskeleton to provide push-off power," in *2014 5th IEEE RAS&EMBS International Conference on Biomedical Robotics and Biomechatronics* (São Paulo), 918–923. doi: 10.1109/BIOROB.2014.6913898

Meng, W., Liu, Q., Zhou, Z., Ai, Q., Sheng, B., and Xie, S. (2015). Recent development of mechanisms and control strategies for robot-assisted lower limb rehabilitation. *Mechatronics* 31, 132–145. doi: 10.1016/j.mechatronics.2015.04.005

Michmizos, K. P., Rossi, S., Castelli, E., Cappa, P., and Krebs, H. I. (2015). Robot-aided neurorehabilitaion: a pediatric robot for ankle rehabilitation. *IEEE Trans. Neural Syst. Rehabil. Eng.* 23. doi: 10.1109/TNSRE.2015.2410773

Nomura, K., Yonezawa, T., Ogitsu, T., Mizoguchi, H., and Takemura, H. (2015). "Development of stewart platform type ankle-foot device for trip prevention support," in *2015 37th Annual International Conference of the IEEE Engineering in Medicine & Biology Society* (Milan), 4808–4811. doi: 10.1109/EMBC.2015.7319469

Oymagil, A. M., Hitt, J. K., Sugar, T., and Fleeger, J. (2008). "Control of a regenerative braking powered ankle foot orthosis," in *Proceedings of the 2007 IEEE 10th International Conference on Rehabilitation Robotics* (Noordwijk), 28–34. doi: 10.1109/ICORR.2007.4428402

Park, Y. L., Chen, B. R., Perez-Arancibia, N. O., Young, D., Stirling, L., Wood, R. J., et al. (2014). Design and control of a bio-inspired soft wearable robotic device for ankle-foot rehabilitation. *Bioinspir. Biomim.* 9:016007. doi: 10.1088/1748-3182/9/1/016007

Ren, Y., Wu, Y., Yang, C., Xu, T., Harvey, R. L., and Zhang, L. (2017). Developing a wearable ankle rehabilitation robotic device for in-bed acute stroke rehabilitation. *IEEE Trans. Neural Syst. Rehabil. Eng.* 25, 589–596. doi: 10.1109/TNSRE.2016.2584003

Roy, A., Krebs,. H. I., Williams, D. J., Bever, C. T., Forrester, L. W., Macko, R. M., et al. (2009). Robot-aided neurorehabilitation: a novel robot for ankle rehabilitation. *IEEE Trans. Robot.* 25, 569–582. doi: 10.1109/TRO.2009.2019783

Saglia, J., Tsagarakis, N., Dai, J. S., and Caldwell, D. (2009). A high-performance redundantly actuated parallel mechanism for ankle rehabilitation. *Int. J. Robot. Res.* 28, 1216–1227. doi: 10.1177/0278364909104221

Sawicki, G. S., and Ferris, D. P. (2009). A pneumatically powered knee-ankle-foot orthosis (KAFO) with myoelectric activation and inhibition. *J. Neuroeng. Rehabil.* 6:23. doi: 10.1186/1743-0003-6-23

Shorter, K. A., Kogler, G. F., Loth, E., Durfee, W. K., and Hsiao-Wecksler, E. T. (2011). A portable powered ankle-foot orthosis for rehabilitation. *J. Rehabil. Res. Dev.* 48, 459–472. doi: 10.1682/JRRD.2010.04.0054

Siegler, S., Chen, J., and Schneck, C. (1988). The three-dimensional kinematics and flexibility characteristics of the human ankle and subtalar joints- part i: kinematics. *J. Biomech. Eng.* 110, 364–373. doi: 10.1115/1.3108455

Takemura, H., Onodera, T., Ding, M., and Mizoguchi, H. (2012). Design and control of a wearable steward platform-type ankle-foot assistive device. *Int. J. Adv. Robot. Syst.* 9:52449. doi: 10.5772/52449

Tsoi, Y., Xie, S. Q., and Graham, A. (2009). Design, modeling and control of an ankle rehabilitation robot. *Des. Control Intell. Robot. Syst.* 177, 377–399. doi: 10.1007/978-3-540-89933-4_18

Wang, C., Fang, Y., Guo, S., and Chen, Y. (2013). Design and kinematical performance analysis of a 3-RUS/RRR redundantly actuated parallel mechanism for ankle rehabilitation. *J. Mech. Robot.* 5:041003. doi: 10.1115/1.4024736

Wang, C., Fang, Y., Guo, S., and Zhou, C. (2015). Design and kinematic analysis of redundantly actuated parallel mechanisms for ankle rehabilitation. *Robotica* 33, 366–384. doi: 10.1017/s0263574714000241

Ward, J., Sugar, T., Boehler, A., Standeven, J., and Engsberg, J. R. (2011). Stroke survivors' gait adaptations to a powered ankle-foot orthosis. *Adv. Robot.* 25, 1879–1901. doi: 10.1163/016918611X588907

Witte, K. A., Zhang, J., Jackson, R. W., and Collins, S. H. (2015). "Design of two lightweight, high-bandwidth torque-controlled ankle exoskeletons," in *2015 IEEE International Conference on Robotics and Automation (ICRA)* (Washington, DC), 1223–1228. doi: 10.1109/ICRA.2015.7139347

Wu, C., Liu, X., Wang, L., and Wang, J. (2010). Optimal design of spherical 5R parallel manipulators considering the motion/force transmissibility. *J. Mech. Design* 132:031003. doi: 10.1115/1.4001129

Wu, C., Liu, X., Wang, L., and Wang, J. (2011). Dimension optimization of an orientation fine-tuning manipulator for segment assembly robots in shield tunneling machines. *Autom. Constr.* 20, 353–359. doi: 10.1016/j.autcon.2010.11.005

Wu, J., Li, T., Wang, J., and Wang, L. (2013). Performance analysis and comparison of planar 3-DOF parallel manipulators with one and two additional branches. *J. Intell. Robot. Syst.* 72, 73–82. doi: 10.1007/s10846-013-9824-8

Xie, F., Liu, X., Wang, L., and Wang, J. (2010). Optimal design and development of a decoupled A/B-axis tool head with parallel kinematics. *Adv. Mech. Eng.* 2, 1652–1660. doi: 10.1155/2010/474602

Yao, S., Zhuang, Y., Li, Z., and Song, R. (2018). Adaptive admittance control for an ankle exoskeleton using an EMG-Driven musculoskeletal model. *Front. Neurorobot.* 12:16. doi: 10.3389/fnbot.2018.00016

Yoshizawa, N. (2010). "Prototype active AFO with ankle joint brake for achilles tendon ruptures," in *2010 3rd International Conference on Biomedical Engineering and Informatics* (Yantai), 1775–1778. doi: 10.1109/BMEI.2010.5639889

Zanganeh, K., and Angeles, J. (1997). Kinematic isotropy and the optimum design of parallel manipulators. *Int. J. Robot. Res.* 16, 185–197. doi: 10.1177/027836499701600205

Zhang, M., Cao, J., Zhu, G., Miao, Q., Zeng, X., and Xie, S. Q. (2017). Reconfigurable workspace and torque capacity of a compliant ankle rehabilitation robot (CARR). *Robot Auton. Syst.* 98, 213–221. doi: 10.1016/j.robot.2017.06.006

Zhou, Z., Zhou, Y., Wang, N., Gao, F., Wei, K., and Wang, Q. (2014). A proprioceptive neuromuscular facilitation integrated robotic ankle-foot system for post stroke rehabilitation. *Robot. Auton. Syst.* 73, 111–122. doi: 10.1016/j.robot.2014.09.023

Zhu, J., Wang, Q., Huang, Y., and Wang, L.ong. (2011). "Adding compliant joints and segmented foot to bio-inspired below-knee exoskeleton," in *2011th IEEE International Conference on Robotics and Automation (ICRA)* (Shanghai), 605–610. doi: 10.1109/ICRA.2011.5980054

Development of a Novel Robotic Rehabilitation System With Muscle-to-Muscle Interface

Jae Hwan Bong[1], Suhun Jung[1], Namji Park[2], Seung-Jong Kim[3] and Shinsuk Park[1]*

[1] Department of Mechanical Engineering, Korea University, Seoul, South Korea, [2] School of Dentistry, Seoul National University, Seoul, South Korea, [3] College of Medicine, Korea University, Seoul, South Korea

In this study, we developed a novel robotic system with a muscle-to-muscle interface to enhance rehabilitation of post-stroke patients. The developed robotic rehabilitation system was designed to provide patients with stage appropriate physical rehabilitation exercise and muscular stimulation. Unlike the position-based control of conventional bimanual robotic therapies, the developed system stimulates the activities of the target muscles, as well as the joint movements of the paretic limb. The robot-assisted motion and the electrical stimulation on the muscles of the paretic side are controlled by on-line comparison of the motion and the muscle activities between the paretic and unaffected sides. With the developed system, the rehabilitation exercise can be customized and modulated depending on the patient's stage of motor recovery after stroke. The system can be operated in three different modes allowing both passive and active exercises. The effectiveness of the developed system was verified with healthy human subjects, where the subjects were paired to serve as the unaffected side and the paretic side of a hemiplegic patient.

Keywords: wearable robot, human-human interface, electromyogram, functional electrical stimulation, lower limb rehabilitation

INTRODUCTION

A large number of patients suffer from lower limb hemiplegia after experiencing a stroke. Post-stroke hemiplegic patients have impaired gait pattern and must undergo rehabilitation exercises to restore their normal gait pattern. Intensive rehabilitation exercises must be conducted within a golden period, between 3 and 6 months following the stroke within which most functional restoration takes place. However, rehabilitation processes often do not begin in a timely manner due to the limited number of therapists available to conduct the exercises. The process of rehabilitation requires significant time and effort for the therapists and therefore the number and duration of rehabilitation sessions hardly meet the demand. It is crucial that this problem be addressed as studies have indicated that increasing the amount of physiotherapy has a positive effect on functional recovery (Kwakkel et al., 2008; Huang and Krakauer, 2009; Kollen et al., 2009; Kwakkel, 2009; Marchal-Crespo and Reinkensmeyer, 2009).

In order to reduce the therapists' workload and thus increase the patients' accessibility of rehabilitation sessions, multiple kinds of robotic rehabilitation systems have been developed. However, some studies have raised concerns regarding robot-assisted rehabilitation systems due to the patient's passivity in conducting the exercise (Israel et al., 2006; Hornby et al., 2008;

Hidler et al., 2009). The degree of functional recovery during rehabilitation depends on the level of task difficulty and the amount of exercise actively conducted by the patient. In order to maximize the efficacy of rehabilitation exercises and thus functional recovery, patients must actively contract their appropriate muscles rather than passively depend on the robot to conduct the pre-programed motions (Hornby et al., 2008; Hidler et al., 2009). Studies have shown that low patient involvement and ease of exercise compromise the speed and outcome of functional recovery (Israel et al., 2006). Leaving the patient idle during the rehabilitation exercise risks wasting the golden period. In order to effectively make use of the golden period, patient involvement in the rehabilitation exercise must be maximized in a timely manner. Passive and active rehabilitation methods should be selected depending on the patient's recovery phases. In the early stage of rehabilitation, the passive exercise is essential to provide the reference trajectories of the motion to patients in order to improve the movability and to reduce muscle atrophy (Jamwal et al., 2014). After recovering a certain degree of muscle strength, the active exercise is necessary to encourage voluntary muscle activation by the patient.

There are three types of control modes that are commonly used for robot-assisted rehabilitation: passive mode, active assist mode, and active resist mode (Marchal-Crespo and Reinkensmeyer, 2009; Pittaccio and Viscuso, 2011). In the passive mode, the patient solely depends on the robot movements that follow the reference trajectories generated by using a position-based trajectory tracking control method (Emken et al., 2008; Beyl et al., 2009; Vallery et al., 2009; Duschau-Wicke et al., 2010; Saglia et al., 2012; Hussain et al., 2013b; Jamwal et al., 2014). The reference trajectories are generated from the movements of the unaffected limb as in bimanual rehabilitation (Lum et al., 2004). Bimanual rehabilitation is a treatment method, in which the patient moves both paretic and unaffected limbs simultaneously. It has been reported that training both limbs in identical motion aids recovery by coupling symmetric proprioceptive feedback in both sides of the ipsilateral corticospinal pathway (Wolf et al., 1989; Burgar et al., 2000). In the active assist mode, the robot provides partial assistance to the patients who recovered muscle strength to produce a voluntary motion. The active resist mode is used to help strengthen the muscle forces of the patients by performing the exercise against a resistive force exerted by the robot (Poli et al., 2013). A number of rehabilitation robots employ the well-known impedance control strategy for the active assist mode and the active resist mode to encourage active participation of the patient and to adjust the dynamic relationship between robot position and contact force (Emken and Reinkensmeyer, 2005; Agrawal et al., 2007; Veneman et al., 2007; Wolbrecht et al., 2008; Roy et al., 2009; Duschau-Wicke et al., 2010; Hussain et al., 2013a; Koopman et al., 2013). In active operation modes with impedance control, it is difficult to stimulate and control the contraction of the specific target muscles that are necessary to generate the movement. In their meta-analyses for the effects of robot-assisted rehabilitation, Veerbeek et al. (2017) reported that robot-assisted rehabilitation can improve motor control ability and muscle strength in the paretic side, while the improvement does not

appear significant. This issue may be alleviated with the aid of functional electrical stimulation (FES) that delivers low intensity electrical stimulation to a specific nerve or muscle to induce muscle contraction artificially. FES is known to be beneficial for improving motor ability and inducing changes in motor cortex excitability and functional cortical reorganization (Maffiuletti et al., 2011; Popović, 2014).

In this study, we aim to develop a robot-assisted rehabilitation system combined with the application of FES on the muscles in the lower limb to enhance the recovery process for hemiplegic stroke patients. Several studies have attempted to provide robotic rehabilitation therapies by using hybrid robotic rehabilitation systems (HRRS), where FES is applied in addition to volitional muscle contraction in order to induce further muscle contraction and thus muscle forces (Langzam et al., 2006a,b; Bulea et al., 2013; Chen et al., 2013). In those studies, the intensity of FES is controlled either by predefined stimulation pattern (Bulea et al., 2013) or by feedback control (Chen et al., 2013), which takes into account of the states of the paretic side only. The robotic system developed in this study employs electromyography (EMG) biofeedback signals from both unaffected and paretic sides to control the motion and muscle activities of the paretic limb. Unlike the position-based control of conventional bimanual robotic therapies, this feature aims to exploit the functional bimanual synergies at the level of muscle activities, as well as at the level of joint movements.

The HRRS developed in this study provides the patient with the passive and active exercises. During the passive exercise, a one-DOF rehabilitation orthosis for knee movement is controlled by a proportional-derivative (PD) controller to provide isokinetic exercise for the paretic leg. The desired position of the paretic leg is set to be the position of the unaffected leg, while the desired velocity is set constant. During the active exercise, both the orthosis motion and the FES intensity on the paretic side are controlled. The orthosis is controlled by admittance controller to generate a target interaction force between the orthosis and the paretic leg. FES applied to the paretic leg is modulated to generate appropriate muscle contraction to follow the knee joint motion of the unaffected leg. FES intensity is controlled by comparing the muscle activities of the paretic and unaffected legs. EMG is measured from Rectus Femoris (RF) – one of the knee extensor muscles. The measured EMG signals from both paretic and unaffected sides are processed and compared to modulate the FES intensity to induce the muscle activity for the RF on the paretic side close to that on the unaffected side. The passive and active operation modes can be selected depending on the patient's stage of motor recovery after stroke. It was reported that the functional restoration of knee extensor muscles, such as RF, plays an important role in regulating comfort and gait speed of hemiplegic patients (Hsu et al., 2003).

This paper is organized as follows: section "Materials and Methods" describes the developed rehabilitation system. Section "Experiments and Results" explains the experimental setup and shows performance evaluation of the developed rehabilitation system. Section "Conclusion" summarizes the major points of the system performance and concludes the study.

MATERIALS AND METHODS

System Overview

The HRRS developed in this study consists of four major components: an exoskeleton robot (ATO, KIST, Seoul, South Korea), sEMG sensors with wireless transmission devices (Trigno Lab, Delsys Inc., United States), a FES device (Rehastim, Hasomed GmbH, Germany), and a self-designed knee angle measurement device.

ATO system developed in the previous study (Lee et al., 2013) is an exoskeleton-type robotic orthosis for one-DOF sagittal knee motion (extension-flexion). The patient's leg is attached to ATO system by a brace at the calf. As shown in **Figure 1**, the joint angle θ is defined 0° when the patient's knee is fully extended along with ATO (**Figure 1A**). While ATO moves in the direction of knee flexion, the joint angle θ is increased as depicted from **Figures 1A–D**. For actuation of the joint angle θ ranging from 0° (**Figure 1A**) to 90° (**Figure 1D**), a linear actuator equipped with a ball screw and a BLDC motor (Maxon EC-4pole 200W) is implemented (see **Figure 2A**). The linear actuator changes the length x in **Figure 2A**, which results in joint angle (θ) regulation. The load cell equipped at the brace measures the interaction force between the patient's leg and ATO (see **Figure 2B**). The patient's leg is strapped tightly to the brace so that the load cell can measure both tensile and compressive interaction forces. The muscle force including volitional portion and FES induced portion is observed by the measured interaction force. For instance, observation of large tensile interaction force means the patient generates the large muscle force in the direction of knee extension. As another example, observation of compressive interaction force same as leg weight means the patient fully relies on ATO and does not generate any muscle force. The measured interaction force is also used to control the sagittal knee motion of ATO through admittance control (see section "Active Assist Mode" and section "Active Resist Mode").
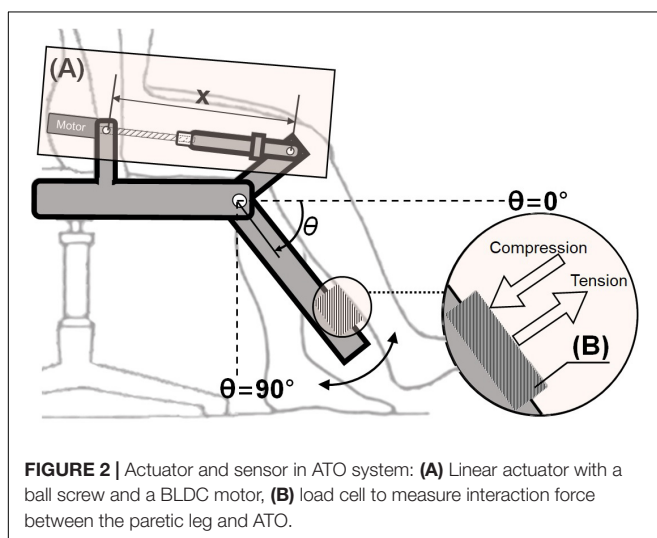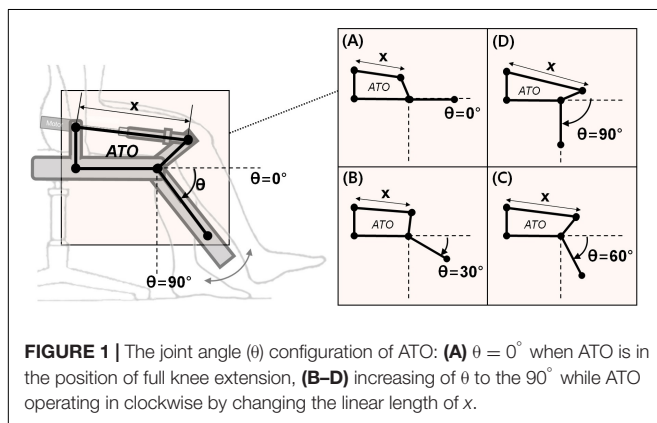
As illustrated in **Figure 3**, the developed system employs two types of interfaces between the unaffected and paretic sides of the patient: muscle-to-muscle Interface and motion-to-motion Interface. The muscle-to-muscle Interface, described in **Figure 3A**, modulates the amplitude of FES on the paretic leg based on the difference in the readings from sEMG sensors on the unaffected and paretic legs. The motion-to-motion Interface, described in **Figure 3B**, controls the sagittal knee motion of ATO and thus guides the knee joint motion of the paretic side based on the knee joint motion of the unaffected side and the interaction force between ATO and the paretic leg. The knee joint motion of the unaffected side was measured by the self-designed knee angle measurement device consisting of a hard-type commercial knee brace and a goniometer. The goniometer was attached on the side of hard-type commercial knee brace to measure the knee motion.

FIGURE 1 | The joint angle (θ) configuration of ATO: **(A)** θ = 0° when ATO is in the position of full knee extension, **(B–D)** increasing of θ to the 90° while ATO operating in clockwise by changing the linear length of x.

FIGURE 2 | Actuator and sensor in ATO system: **(A)** Linear actuator with a ball screw and a BLDC motor, **(B)** load cell to measure interaction force between the paretic leg and ATO.

FIGURE 3 | Interfaces between unaffected and paretic legs: **(A)** Muscle-to-Muscle Interface and **(B)** Motion-to-Motion Interface.

Muscle-to-Muscle Interface

The muscle-to-muscle interface compares the EMG data from the RF on the paretic and unaffected legs to adjust the amplitude of FES applied on the paretic leg. The EMG records electrical activity in a muscle, which reflects degree of muscle activation. The EMG data from the RF on the paretic and unaffected legs are measured and processed. Based on the difference between the two processed EMG readings, the intensity of FES is modulated and applied to the RF on the paretic leg. Through FES, the patients can train to learn their appropriate amount and timing of muscle activation of the RF on the paretic leg.

Since the raw EMG readings from different muscles show different characteristics in amplitude and frequency, it is difficult to compare the raw EMG signals without preprocessing (De Luca, 1997). Moreover, the EMG data from the paretic side contains stimulus artifacts induced by FES. In this study, the EMG signals are processed using filtering, feature extraction, and normalization techniques. The filter was designed using a combination of a blanking window and a comb filter. The blanking window is used for the EMG signal from the paretic side to nullify stimulation artifacts from the first 25 ms of the FES pulse (Frigo et al., 2000). After applying the blanking window, the comb filter removes harmonic artifacts and thus isolates the volitional component of the EMG signal (Frigo et al., 2000). The comb filter is a finite impulse response (FIR) filter and can be expressed as follows:

$$y(t) = \frac{x(t) - x(t - T)}{\sqrt{2}} \tag{1}$$

In equation (1), $x(t)$ and $y(t)$ are the raw EMG and the filtered EMG at time t. T denotes a time period of FES.

Feature extraction technique is applied on the filtered EMG using waveform length, which is effective for extracting time-domain features including waveform amplitude, frequency, and duration (Phinyomark et al., 2010; Negi et al., 2016; Veer and Sharma, 2016). The waveform length can be expressed as follow:

$$y = \sum_{i=0}^{N-1} |x_{i+1} - x_i| \tag{2}$$

In equation (2), x and y are the filtered EMG and the waveform length, respectively. N is a constant related to the number of samples to be used for calculating waveform length. In this paper, N was used for 160.

As a common normalization method, EMG signals are divided by a reference value. The reference value was taken by the maximum EMG value (Halaki and Ginn, 2012). Maximum activation for each subject was obtained beforehand while performing the task under maximum effort.

After the signal is processed according to procedures mentioned above, the difference between the processed EMG signals from RF on the paretic and unaffected legs is calculated. Based on the EMG difference, the amplitude of FES applied on the paretic leg is determined. If the EMG difference is less than 0.01, the FES amplitude is maintained. If the EMG difference is greater than 0.01 and the EMG signal from the RF on the

unaffected leg is larger than that from the RF on the paretic leg, the FES amplitude is increased by 2 mA. If the EMG difference is greater than 0.01 and the EMG signal from the RF on the unaffected leg is smaller than that from the RF on the paretic leg, the FES amplitude is decreased by 2mA. Altogether, the FES amplitude is determined as follows:

$$EMG = EMG_{unaffected} - EMG_{paretic} \tag{3}$$

$$FES_i = \begin{cases} FES_{i-1} + 2 & (EMG > 0.01) \\ FES_{i-1} & (|EMG| \leq 0.01) \\ FES_{i-1} - 2 & (EMG < -0.01) \end{cases} \tag{4}$$

In equation (3), $EMG_{unaffected}$ and $EMG_{paretic}$ indicate the processed EMG signals from RF on the unaffected and paretic legs, respectively. In equation (4), FES_i is the amplitude of ith FES applied to the RF on the paretic leg.

Motion-to-Motion Interface

The motion-to-motion interface maps the knee joint motion of the unaffected leg to the motion of joint movement of ATO worn on the paretic leg. The motion-to-motion interface can provide the patient with three types of control modes: passive mode, active assist mode, and active resist mode. In the passive mode, the interface conducts robotic motion assistance for the patient incapable of generating volitional muscle contractions. While the joint angle (θ) of ATO is controlled by the knee angle of the unaffected leg, the joint velocity of ATO is kept constant during extension and flexion. This type of isokinetic exercise has been reported to be appropriate in the early phase of rehabilitation (Cabri and Clarys, 1991). Two different active modes, the active assist mode and the active resist mode, are combined with the muscle-to-muscle interface described in section "Muscle-to-Muscle Interface." The interface provides assistive or resistive forces to the paretic leg using admittance control based on the knee joint motion of the unaffected leg and the interaction force between the paretic leg and ATO.

Figure 4 shows the block diagram of the admittance controller used for the two active modes. First, the target interaction force between the paretic leg and the exoskeleton is calculated based on the difference between the knee joint angles of the unaffected leg and the joint angle of ATO worn on the paretic leg (**Figure 4A**). The target interaction forces are set differently for the active assist mode and the active resist mode. Detail of setting for the target interaction force in each active mode is described in section "Active Assist Mode" and section "Active Resist Mode." The difference between the target interaction force and the actual interaction force is calculated. Then, the target joint velocity for PD controller is calculated to reduce the difference between the target and actual interaction force (**Figure 4B**). The calculation of the target joint velocity has following expressions:

$$\dot{\theta}_T = f(\Delta F) = \frac{\Delta F}{b} \tag{5}$$

$$\Delta F = IF_T - IF_C \tag{6}$$
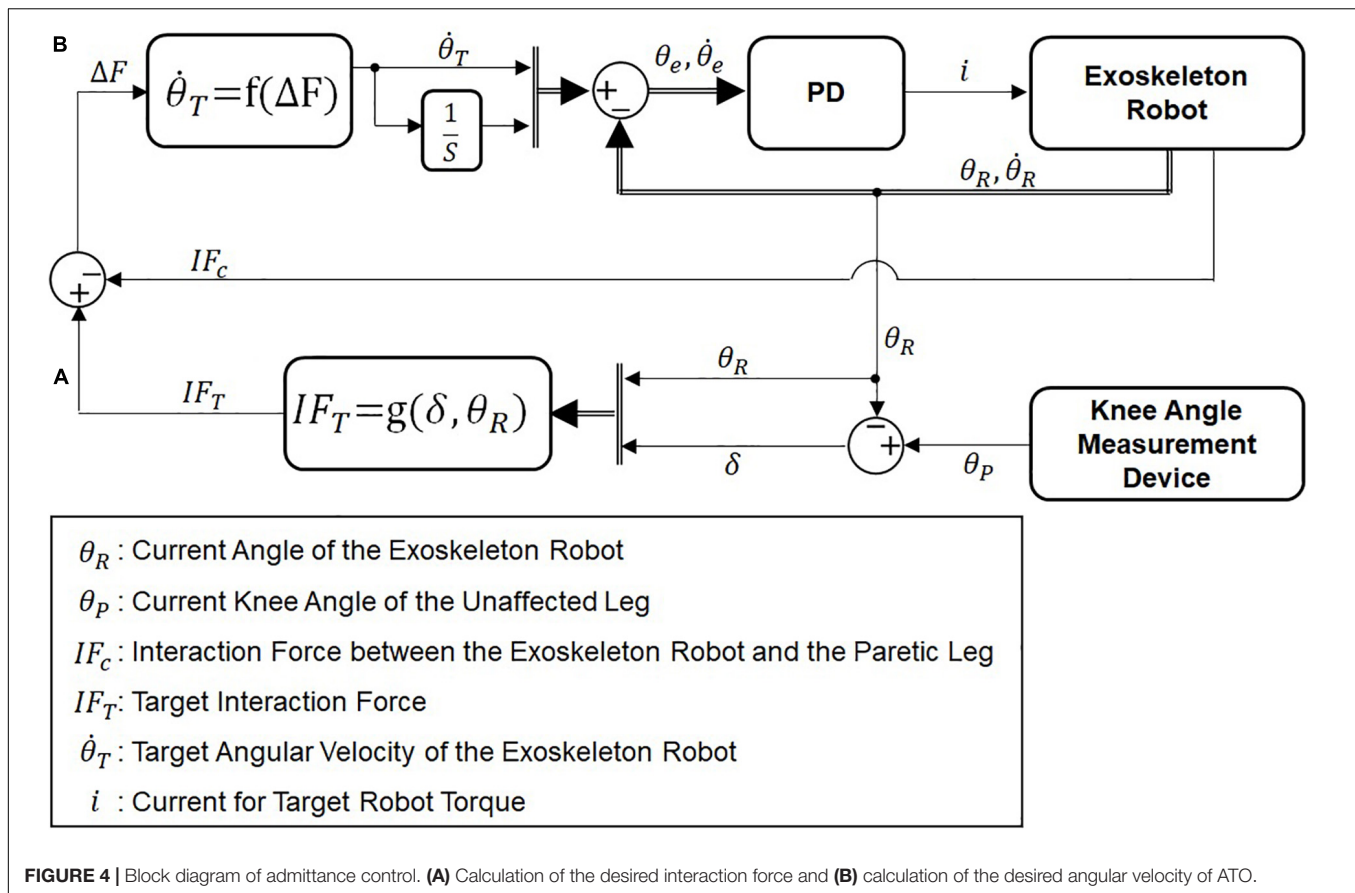
FIGURE 4 | Block diagram of admittance control. **(A)** Calculation of the desired interaction force and **(B)** calculation of the desired angular velocity of ATO.

In equation (5), b is a constant coefficient between the force difference ($\triangle F$) and the target joint velocity ($\dot{\theta}_T$). In this paper, b was used for 30. In equation (6), the force difference ($\triangle F$) is calculated by difference of the target interaction force (IF_T) and the actual interaction force (IF_C). As the force difference increases, the joint velocity increases in the direction of force difference, which in turn will decrease the interaction force and thus the joint velocity.

Operation Modes

The developed HRRS can operate in three different levels of difficulty by combining the motion-to-motion interface and the muscle-to-muscle interface. Three operation modes in the developed HRRS is named after the three different control modes in the motion-to-motion interface. Depending on the patient's stage of motor recovery after stroke (Gowland et al., 1993), the operation mode can be adjusted to provide appropriate amount of support and also encourage maximum involvement by the patient.

Passive Mode

In the passive mode, only the motion-to-motion interface is used to provide isokinetic exercise for the paretic leg. Flexion and extension of the paretic leg are commanded by the motion of the unaffected leg through position-based control. The muscle-to-muscle interface is not used in this mode, since the exoskeleton

robot produces the movements even without volitional muscle contractions in the paretic leg.

Active Assist Mode

In the active assist mode, both the muscle-to-muscle interface and the motion-to-motion interface are used to assist active exercise for the paretic leg through hybrid muscle activation and robotic assistive force.

With the muscle-to-muscle interface, the FES on the paretic leg is modulated by feedback of the EMG signals from both unaffected and paretic legs. The motion-to-motion interface controls the joint displacement and the joint velocity of ATO based on the knee motion of the unaffected leg and the interaction force between the paretic leg and ATO using admittance control.

In this mode, the patient needs to activate one's RF for knee extension, while muscle activation of the RF is not necessary during knee flexion. The muscle-to-muscle interface compensates for the deficiency in muscle activation of RF on the paretic leg. The motion-to-motion interface assists the motion of the paretic leg based on admittance control as described in section "Motion-to-Motion Interface."

Figure 5 plots the target interaction force for admittance control (solid line). The target interaction force is determined based on the gravity compensation force to counterbalance the weight of the paretic leg and ATO worn on the leg (dashed line). In this figure, the compressive force measured by the load cell

FIGURE 5 | Target interaction force for the robotic force assistance in the active assist mode.
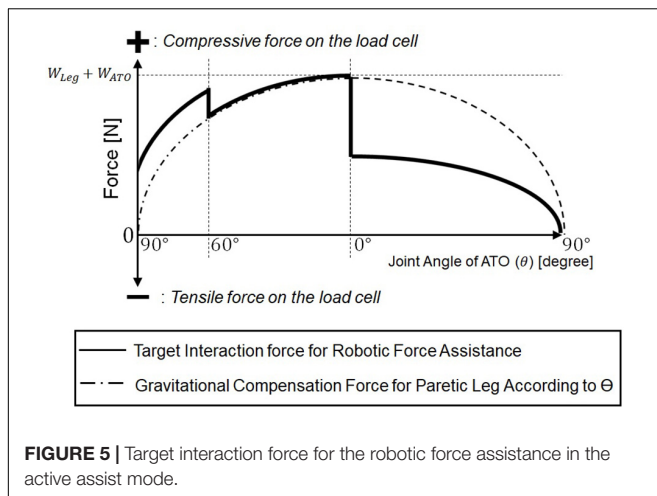
is positive, while the tensile force is negative. In this mode, the target interaction force during whole knee movement is set to be compressive to assist the paretic leg in the direction of knee extension. The gravity compensation force, G (θ), related to the joint angle, θ (see **Figure 1**), has following expression:

$$G(\theta) = (W_{Leg} + W_{ATO}) \cos(\theta) \tag{7}$$

In equation (5), W_{Leg} and W_{ATO} are weights of the paretic leg and the ATO. As shown in the **Figure 5**, the gravity compensation force, G (θ), increases as the joint angle (θ) changes from 90° (**Figure 1D**) to 0° (**Figure 1A**), and it decreases as the joint angle changes from 0° (**Figure 1A**) to 90° (**Figure 1D**). From 90° (**Figure 1D**) to 60° (**Figure 1C**) during knee extension motion, an assistive force larger than the gravity compensation force is applied so that the paretic leg can start the extension motion without any muscle force. From 60° (**Figure 1C**) to 0° (**Figure 1A**) during the rest of knee extension motion, an assistive force equal to the gravity compensation force is applied, and FES is applied to the RF on the paretic leg through the muscle-to-muscle interface in case of muscle activation deficiency in the paretic leg. From 0° (**Figure 1A**) to 90° (**Figure 1D**) during knee flexion, half of the gravity compensation force is applied to prevent excessive joint velocity during knee flexion, which may cause knee injury.

Active Resist Mode

In the active resist mode, both the muscle-to-muscle interface and the motion-to-motion interface are used. With the muscle-to-muscle interface, the FES on the paretic leg is controlled by the difference between the EMG signals from the unaffected leg and the paretic leg.

This mode differs from the active assist mode in that the motion-to-motion interface applies resistive force against the direction of hybrid muscle activation.

In this mode, the patient needs to activate the RF on the paretic leg to overcome the robotic resistive force during both knee extension and knee flexion. This kind of resistance exercise is highly effective for hemiplegia patients in re-gaining muscle strength in their lower limbs (Wist et al., 2016).

The muscle-to-muscle interface compensates for the deficiency in muscle activation of the RF on the paretic leg to overcome the resistive force generated by ATO. During knee extension, the hybrid muscle activation serves for concentric contraction of the RF to overcome the load in the direction of knee flexion. During knee flexion, the hybrid muscle activation serves for eccentric contraction of the RF, while ATO constrains the paretic leg to make knee flexion movement.

In this mode, the target interaction force for the motion-to-motion interface is set to be tensile and constant against the contraction of RF on the paretic leg during knee extension and flexion.

EXPERIMENTS AND RESULTS

Experimental Setup

A total of six healthy subjects aged 25 – 32 participated in this study. The developed system was tested five times, each with two subjects. Each time, one subject (Subject A, B, C, D, or E) served as the paretic side of a hemiplegic patient, while the other subject (Subject G) served as the unaffected side (**Figure 6**). Each pair of subjects performed the exercise with the three operation modes in same order of (1) Passive Mode, (2) Active Assist Mode, and (3) Active Resist Mode. Each pair of subjects performed each operation mode in different 3 days to minimize learning or

FIGURE 6 | Experimental setup with two healthy subjects. A written informed consent for the publication was obtained from the individuals in this image. **(A)** The exoskeleton robot, ATO, **(B)** the electrodes for the FES, **(C)** sEMG sensors, and **(D)** the self-designed knee angle measurement device.
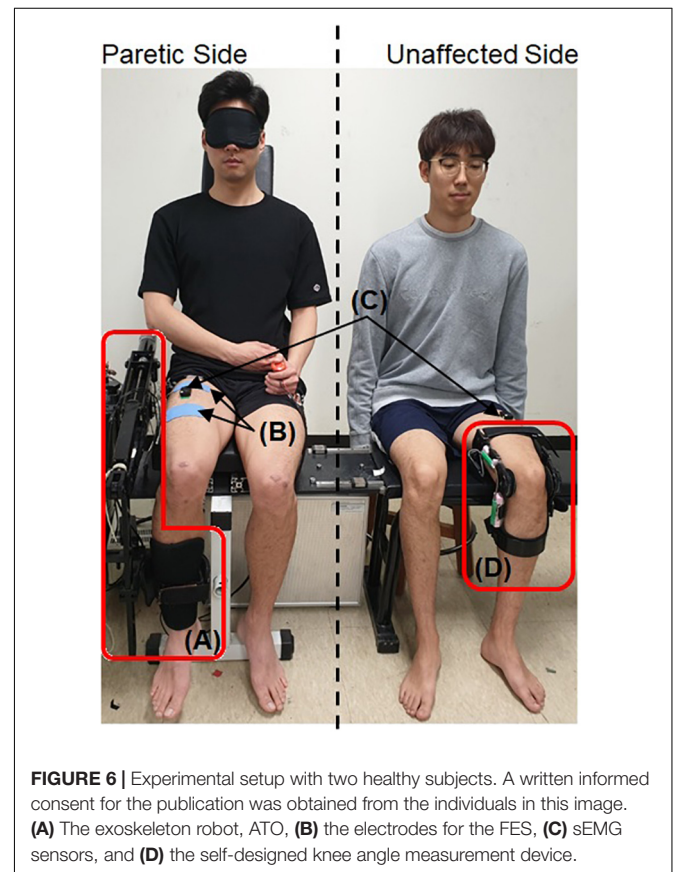
fatigue effect. Thus, each operation mode was tested five times by five different pairs of subjects in different days.

The knee angle measurement device and the sEMG sensor were attached to the subject on the unaffected side. The exoskeleton robot, sEMG sensor and FES electrodes were attached to the subject on the paretic side. The subject on the paretic side was blindfolded to prevent any inadvertent muscle activation caused by observing the movement of the subject on the unaffected side. The maximum amplitude of FES applied on the paretic side was limited to 2mA lower than the pain threshold of each human subject. The range of knee movement for the unaffected side which has the same configuration as the joint angle (θ) of ATO (shown in **Figure 1**) was limited to 5° to 85° for safety.

In the experiment using the Passive Mode, the subject on the paretic side was instructed to neither generate volitional muscle force nor resist the motion of the exoskeleton robot. This is to imitate the state of post-stroke patients in early stages of motor recovery. The subject on the unaffected side was instructed to firstly carry out the knee extension and then to confirm visually that the subject on the paretic side finished the knee extension prior to conducting knee flexion.

In the experiment using the Active Assist Mode, the subject on the paretic side was instructed to generate volitional muscle force only when FES was applied to the RF. The instruction for the subject on the unaffected side was the same as described in the first experiment using the section "Passive Mode."

In case of the Active Resist Mode, for knee extension, the subject on the paretic side was instructed to generate volitional muscle force only when FES was applied to the RF. For knee flexion, the subject was instructed to volitionally contract the muscle with or without FES application. For the subject on the unaffected side, additional weight was attached to the leg, so that the subject can generate a muscle force larger than that in other operation modes. During knee extension, the subject was required to produce a larger muscle force (concentric contraction) to overcome the extra weight. Also, the subject on the unaffected side was instructed to perform the knee flexion slowly. Due to the extra weight, the subject was required to maintain the muscle activation of the RF during knee flexion (eccentric contraction).

The experiments involving human subjects were approved by the Institutional Review Board at Korea University in Seoul, South Korea (KUIRB-2019-0061-01).

Comparison of Experimental Results Among the Operation Modes

Figure 7 shows the results from one experimental session in three different operation modes described in section "Operation Modes": the passive mode (shown in the first column), the active assist mode (shown in the second column), and the active resist mode (shown in the third column).

Experiments Using Passive Mode

The first column in **Figure 7** shows the results from a pair of subjects in the passive mode. Graphs (A) and (B) plot the knee joint angles and velocities on the paretic side (solid line) and on the unaffected side (dashed line), respectively. Graph (C) plots the normalized EMG measurements from the recti femoris on the paretic side (solid line) and the unaffected side (dotted line). The bold line in Graph (C) indicates the EMG difference, which is calculated by subtracting the normalized EMG of the RF on the paretic side from that on the unaffected side. Graph (E) plots the interaction force measured by the load cell. In Graph (A), the knee joint angle of the paretic side follows that of the unaffected side with a small delay. Graph (B) shows that the joint velocity of the paretic side is kept constant owing to the isokinetic control described in section "Passive Mode," while the joint velocity of the unaffected side is controlled arbitrarily by the subject on the unaffected side.

As can be seen in Graph (C), the normalized EMG of the RF on the paretic side is nearly zero, which indicates that the RF on the paretic side is not volitionally activated unlike the unaffected side.

In Graph (E), the interaction force is kept compressive (positive sign), since the exoskeleton robot counterbalances the weight of the leg while assisting the motion.

The results show that the passive mode of developed HRRS assisted passive exercises at constant joint velocity without any volitional muscle activation from the subject on the paretic side.

Experiments Using Active Assist Mode

The second column of **Figure 7** shows the results from a pair of subjects in the active assist mode. Graph (A) plots the knee joint angles on the paretic side (solid line) and on the unaffected side (dashed line). Graph (C) plots the normalized EMG measurements from the recti femoris on the paretic side (solid line) and the unaffected side (dotted line). The bold line in graph (C) indicates the EMG difference. Graph (D) plots the amplitude of FES applied to the RF on the paretic side, which is modulated based on the EMG difference, as described in section "Muscle-to-Muscle Interface." Graph (E) plots the interaction force measured by the load cell. Graph (A) shows that the knee joint angle of the paretic side follows that of the unaffected side with a considerable time delay. The time delay mainly results from the mechanism of the admittance controller, which tracks the target interaction force rather than the target joint angle. During knee extension [shaded box (1)] from 60° (see **Figure 1C**) to 0° (see **Figure 1A**), additional muscle force is required to reach full extension (joint angle 0° depicted in **Figure 1A**), since only the gravitational force is compensated by the exoskeleton robot. As can be seen in the **Figure 7** [marked by the arrow in graph (A)], the leg motion stops and waits until additional muscle force is provided by FES.

During knee extension [shaded box (1)], the RF on the unaffected side is activated, and thus the EMG difference increases over the threshold of 0.01, which in turn raises the FES amplitude as shown in graphs (C) and (D). The RF on the paretic side is activated by hybrid muscle activation from both volitional muscle contraction and FES. As the knee flexion [shaded box (2)] starts, the RF on the unaffected side is deactivated, and the EMG difference decreases under the threshold of −0.01, which in turn lowers the FES amplitude. The EMG difference drops to the value between −0.01 and 0.01 since the RF on both sides become deactivated. During knee flexion, the FES amplitude applied to

FIGURE 7 | Experimental results from three operation modes: the passive mode in the first column, the active assist mode in the second column, and the active resist mode in the third column. **(A)** Knee joint angle, **(B)** knee joint velocity, **(C)** normalized EMG, **(D)** FES amplitude, and **(E)** interaction force (positive for compression, negative for tension). Shading boxes (1) and (2) indicate the knee extension (from 90° to 0°) and flexion (from 0° to 90°), respectively.

the RF on the paretic side remains nearly zero with no hybrid muscle activation on the paretic side.

During knee extension [shaded box (1)] from 90° (see **Figure 1D**) to 60° (see **Figure 1C**), the interaction force is compressive (positive sign) because the leg motion of the paretic side is supported by ATO as shown in graph (E). The EMG [solid line in graph (C)] and the FES amplitude [graph (D)] on the paretic side are nearly zero, which indicates that the hybrid

muscle activation is not applied in this range. When the FES is applied at around joint angle 60° (see **Figure 1C**), the RF on the paretic side becomes activated. This muscle activation accelerates the knee extension motion causing the leg motion to overshoot the motion of ATO. In Graph (E), the tensile interaction force (negative sign) is caused by this overshoot [shaded box (1)]. During knee flexion [shaded box (2)], the interaction force is compressive (positive sign), and the normalized EMG of the RF
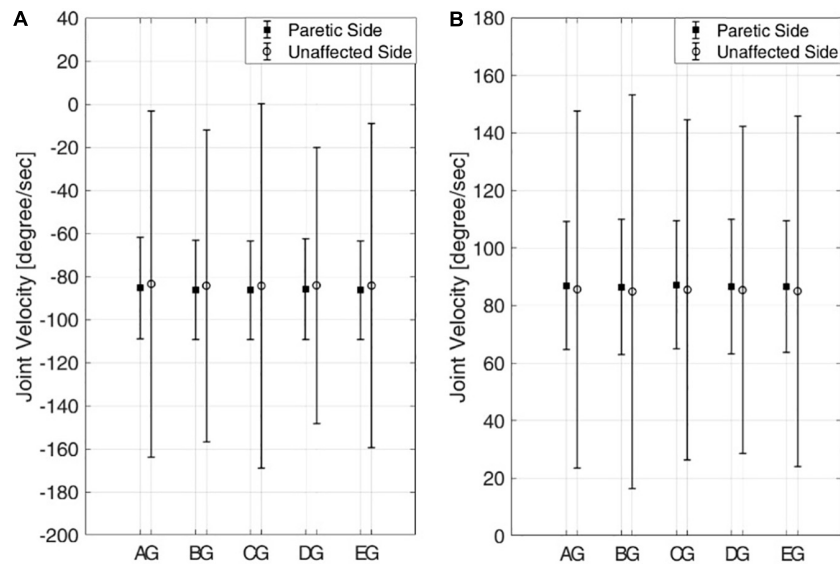
FIGURE 8 | Means and standard deviations of knee joint velocities from five pairs of six subjects in the passive mode. **(A)** Knee extension and **(B)** knee flexion. Six subjects are denoted by A–E,G.

on the paretic side is close to zero, because the weight of the leg is supported by the exoskeleton robot and muscle activation to counterbalance the weight is unnecessary during flexion.

Experiments Using Active Resist Mode

The graphs in the third column of **Figure 7** show the results from a pair of subjects in the active resist mode. Graph (A) plots the knee joint angles on the paretic side (solid line) and on the unaffected side (dashed line). Graph (C) plots the normalized EMG measurements from the RF on the paretic side (solid line) and the unaffected side (dotted line). The bold line in graph (C) indicates the EMG difference. Graph (D) plots the amplitude of FES applied to the RF on the paretic side. Graph (E) plots the interaction force measured by the load cell. Graph (A) shows that the knee joint angle of the paretic side follows that of the unaffected side with a considerable time delay, as does in the active assist mode (section "Experiments Using Active Assist Mode"). As can be seen in the **Figure 7**, the leg on the paretic side did not reach full extension (joint angle 0° depicted in **Figure 1A**) because even the maximum amount of hybrid muscle activation was not able to overcome the robotic force resistance applied by ATO. As the knee extension starts [shaded box (1)], the RF on the unaffected side produces a large force owing to the extra weight imposed on the unaffected leg. This additional load on the unaffected leg, in turn, increases the FES amplitude on the paretic leg to its maximum value. The hybrid muscle activation from volitional muscle contraction and FES activates the RF on the paretic side during knee extension. Unlike in other operation modes, the muscle on the paretic side is highly activated without FES application during knee flexion [shaded box (2)]. As can be seen in Graphs (C) and (D), the EMG difference shows negative values during knee flexion with nearly zero FES amplitude, which suggests that the RF on the paretic side is more activated than that

on the unaffected side to resist the robotic force applied in the direction of knee flexion.

As can be seen in graph (E), the interaction force shows large negative values (tensile force) for both knee extension and knee flexion. The large tensile force is caused by the hybrid muscle activation during the knee extension and the volitional muscle activation during the knee flexion.

Comparison of Experimental Results Among the Subjects

Figure 8 shows the means and standard deviations of the knee joint velocities from the five pairs of six subjects during knee extension (**Figure 8A**) and knee flexion (**Figure 8B**) in the passive mode. While the mean velocities from the paretic and unaffected sides are nearly the same, the standard deviations of the joint velocities from the paretic side are much lower than those from the unaffected side for all the pairs of subjects for both knee extension and flexion.

These results indicate that joint motion on the paretic side is kept isokinetic even if the joint motion on the unaffected side, which commands the motion on the paretic side, shows varying joint velocity. Also, the joint velocities of the subjects on the paretic side show no statistically significant differences across the subjects for both extension and flexion (one-way ANOVA with Turkey-Kramer *post hoc* analysis, $p = 0.71 > 0.05$ for knee extension and $p = 0.74 > 0.05$ for knee flexion).

Figure 9 shows the min-mean-max values of the interaction force for five subjects on the paretic side during knee extension (**Figure 9A**) and knee flexion (**Figure 9B**) in the passive mode (triangle), the active assist mode (square), and the active resist mode (circle).

In the passive mode, the interaction forces are compressive (positive sign) for both knee extension and flexion. This indicates
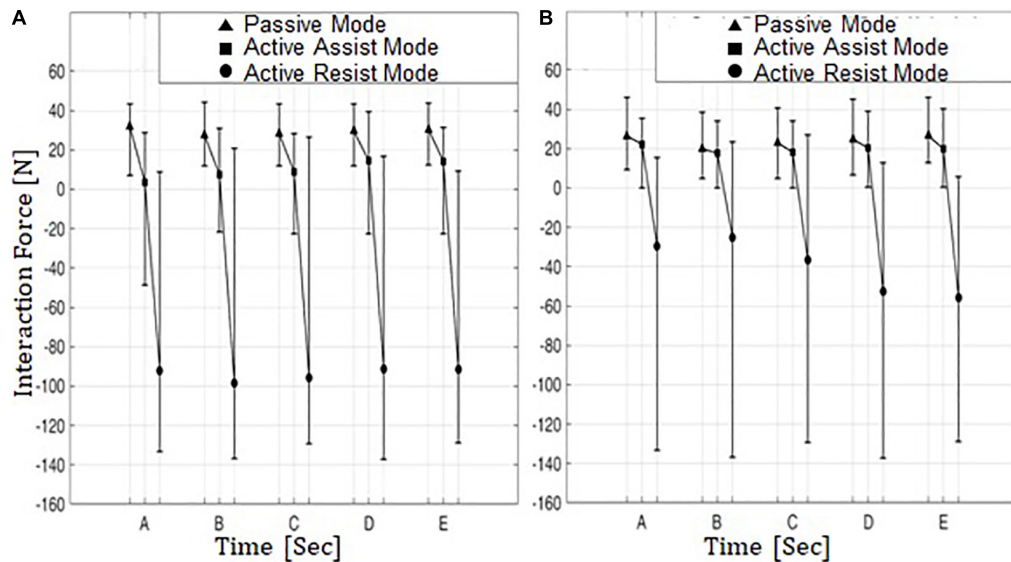
FIGURE 9 | Min-Mean-Max values of interaction force from five subjects on the paretic side in three operation modes. **(A)** Knee extension and **(B)** knee flexion. Five subjects are denoted by A–E.
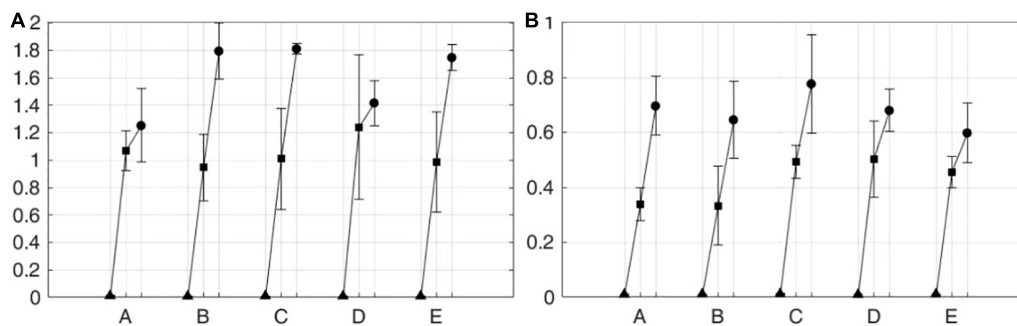
FIGURE 10 | iEMG from five subjects on the paretic side in three operation modes. **(A)** Knee extension and **(B)** knee flexion. Five subjects are denoted by A–E.

that in this operation mode the subjects on the paretic side were passively engaged in knee movement and were relying on robotic motion assistance for conducting the movements.

In the active assist mode, the interaction forces range from compressive (positive sign) to tensile (negative sign) values during knee extension, while the mean interaction forces are slightly compressive. The tensile interaction force can be interpreted as the leg motion outpacing the motion of ATO. This indicates that hybrid muscle contraction plays a considerable role during knee movement with the aid of ATO in the direction of knee extension. During knee flexion, however, the interaction forces are compressive (positive sign), and the mean interaction forces are lower than those in the passive assist mode. This shows that the subjects on the paretic side did not solely rely on ATO during knee flexion, while using the hybrid muscle activation to partially counterbalance the weights of their legs.

In the active resist mode, the interaction forces were mostly in the negative (tensile) range with large negative means for both knee extension and flexion. It appears that in this operation

mode the hybrid muscle contraction plays a much larger role in knee movement compared to the other two operation modes. This indicates that the subjects on the paretic side were more actively engaged in the knee movements to overcome the resistive robotic force.

Figure 10 shows the means and standard deviations of integrated EMG (iEMG) for five subjects on the paretic side during knee extension (**Figure 10A**) and knee flexion (**Figure 10B**) in the passive mode (triangle), the active assist mode (square), and the active resist mode (circle). The iEMG is the time integral of the EMG signal, and it is reported to represent the volitional component of muscle force (Metral and Cassar, 1981). For both knee extension and flexion, iEMG has the maximum mean values in the active resist mode with robotic force resistance (circle) and the minimum mean values in the passive mode with robotic motion assistance (triangle).

Table 1 lists the p-values from t-tests among different operation modes for each subject on the paretic side. As shown in the table, for all the subjects, statically significant differences

TABLE 1 | T-test results among different operation modes.

Comparing group pairs	Subject	Knee movement	
		Knee extension	Knee flexion
Passive mode V.S. Active assist mode	A	8.9E-05*	0.00027*
	B	0.0045*	0.0073*
	C	0.0039*	6.5E-05*
	D	0.0064*	0.0013*
	E	0.0039*	0.0014*
Passive mode V.S. Active resist mode	A	2.5E-05*	0.00014*
	B	0.0043*	0.0037*
	C	0.00016*	0.00066*
	D	4.3E-05*	4.5E-05*
	E	1.9E-06*	0.0014
Active assist mode V.S. Active resist mode	A	0.29	0.00036*
	B	0.0076*	0.27
	C	5.2E-08*	0.049*
	D	0.51	0.042*
	E	0.0063*	0.04*

*Statistically significant difference with $p < 0.05$.

were observed in iEMG between the passive mode and the two active modes for both knee extension and flexion. Between the two active modes, the iEMG values from all the subjects except for subjects A and D show statistically significant differences for knee extension. For knee flexion, the differences in the iEMG values were statically significant between the two active modes for all the subjects except for subject B.

These results show that volitional muscle activity or active engagement of the subjects on the paretic side can be effectively controlled by the operation modes of the developed HRRS.

CONCLUSION

In this study, we developed an exoskeleton-type robotic rehabilitation system for post-stroke patients. For proprioceptive feedback from the unaffected side to the paretic side, the developed robotic system is equipped with two types of interfaces: muscle-to-muscle interface, and motion-to-motion interface.

Unlike the position-based control of conventional bimanual robotic therapies, the developed system is capable of simultaneously stimulating the muscle activities and the joint movements of the paretic limb. Using biofeedback of EMG and functional electric stimulation (FES), the developed rehabilitation system was designed to provide patients with appropriate muscular stimulation considering their stage of motor recovery after stroke. Based on the patient's condition, the system can be operated in three modes with varying levels of difficulty: the passive mode, the active assist mode and the active resist mode.

The effectiveness of the developed HRRS was tested with five different pairs of healthy human subjects, where one of the two subjects participated as the unaffected side and the other as the paretic side of a hemiplegic patient.

Through repetition of rehabilitation exercises with the developed system, patients can naturally learn the timings at which different muscle groups should be activated to make a joint movement. The methodology developed in this study can be extended to multi-joint rehabilitation systems, such as gait rehabilitation and upper limb rehabilitation.

Further studies are required for clinical application of the developed system. The real-time FES artifact removal technique used in this study needs to be refined to accurately extract volitional components of muscle activity. The efficacy of the developed rehabilitation system should be evaluated with a larger number of post-stroke patients in a clinical setup under supervision of rehabilitation medicine physicians.

DATA AVAILABILITY STATEMENT

All datasets generated for this study are included in the article/supplementary material.

ETHICS STATEMENT

The studies involving human participants were reviewed and approved by the Institutional Review Board at Korea University in Seoul, South Korea (KUIRB-2019-0061-01). The patients/participants provided their written informed consent to participate in this study. Written informed consent was obtained from the individual(s) for the publication of any potentially identifiable images or data included in this article.

AUTHOR CONTRIBUTIONS

JB and SP conceived and designed the interfaces, the muscle-to-muscle interface, the motion-to-motion interface, and merged the interfaces and made three operation modes concerning the rehabilitation stages. S-JK designed and realized the rehabilitation orthosis, ATO, and used for a test bed of the interfaces. JB, SJ, and NP performed the experiments and analyzed the experimental data. Finally, JB wrote the manuscript with the help of SP.

FUNDING

REFERENCES

Agrawal, S. K., Banala, S. K., Fattah, A., Sangwan, V., Krishnamoorthy, V., Scholz, J. P., et al. (2007). Assessment of motion of a swing leg and gait rehabilitation with a gravity balancing exoskeleton. *IEEE Trans. Neural. Syst. Rehabil. Eng.* 15, 410–420. doi: 10.1109/tnsre.2007.903930

Beyl, P., van Damme, M., van Ham, R., Vanderborght, B., and Lefeber, D. (2009). Design and control of a lower limb exoskeleton for robot-assisted gait training. *Appl. Bionics Biomech.* 6, 229–243. doi: 10.1080/11762320902784393

Bulea, T. C., Kobetic, R., Audu, M. L., and Triolo, R. J. (2013). Stance controlled knee flexion improves stimulation driven walking after spinal cord injury. *J. Neuroeng. Rehabil.* 10:68. doi: 10.1186/1743-0003-10-68

Burgar, C. G., Lum, P. S., Shor, P. C., and Van der Loos, H. F. M. (2000). Development of robots for rehabilitation therapy: the palo alto VA/Stanford experience. *J. Rehabil. Res. Dev.* 37, 663–673.

Cabri, J. M. H., and Clarys, J. P. (1991). Isokinetic exercise in rehabilitation. *Appl. Ergon.* 22, 295–298. doi: 10.1016/0003-6870(91)90383-s

Chen, Y., Hu, J., Wang, W., Peng, L., and Hou, Z. G. (2013). "An FES-assisted training strategy combined with impedance control for a lower limb rehabilitation robot," in *Proceedings of the IEEE International Conference on Robotics and Biomimetics (ROBIO)*, Shenzhen.

De Luca, C. J. (1997). The use of surface electromyography in biomechanics. *J. Appl. Biomech.* 13, 135–163. doi: 10.1123/jab.13.2.135

Duschau-Wicke, A., von Zitzewitz, J. J., Caprez, A., Luenenburger, L., and Riener, R. (2010). Path control: a method for patient-cooperative robot-aided gait rehabilitation. *IEEE Trans. Neural. Syst. Rehabil. Eng.* 18, 38–48. doi: 10.1109/TNSRE.2009.2033061

Emken, J. L., Harkema, S. J., Beres-Jones, J. A., Ferreira, C. K., and Reinkensmeyer, D. J. (2008). Feasibility of manual teach-and-replay and continuous impedance shaping for robotic locomotor training following spinal cord injury. *IEEE Trans. Biomed. Eng.* 55, 322–334. doi: 10.1109/TBME.2007.910683

Emken, J. L., and Reinkensmeyer, D. J. (2005). Robot-enhanced motor learning: accelerating internal model formation during locomotion by transient dynamic amplification. *IEEE Trans. Neural. Syst. Rehabil. Eng.* 13, 33–39. doi: 10.1109/tnsre.2004.843173

Frigo, C., Ferrarin, M., Frasson, W., Paven, E., and Thorsen, R. (2000). EMG signals detection and processing for on-line control of functional electrical stimulation. *J. Electromyogr. Kinesiol.* 10, 351–360. doi: 10.1016/s1050-6411(00)00026-2

Gowland, C., Stratford, P., Ward, M., Moreland, J., Torresin, W., van Hullenaar, S., et al. (1993). Measuring physical impairment and disability with the Chedoke-McMaster stroke assessment. *Stroke* 24, 58–63. doi: 10.1161/01.str.24.1.58

Halaki, M., and Ginn, K. (2012). *Normalization of EMG Signals: To Normalize or Not to Normalize and What to Normalize to? in Computational Intelligence in Electromyography Analysis*. London: INTECH Open Access Publisher.

Hidler, J., Nichols, D., Pelliccio, M., Brady, K., Campbell, D. D., Kahn, J. H., et al. (2009). Multicenter randomized clinical trial evaluating the effectiveness of the lokomat in subacute stroke. *Neurorehabil. Neural Repair* 23, 5–13. doi: 10.1177/1545968308326632

Hornby, T. G., Campbell, D. D., Kahn, J. H., Demott, T., Moore, J. L., and Roth, H. R. (2008). Enhanced gait-related improvements after therapist-versus robotic-assisted locomotor training in subjects with chronic stroke - A randomized controlled study. *Stroke* 39, 1786–1792. doi: 10.1161/STROKEAHA.107.504779

Hsu, A. L., Tang, P. F., and Jan, M. H. (2003). Analysis of impairments influencing gait velocity and asymmetry of hemiplegic patients after mild to moderate stroke. *Arch. Phys. Med. Rehabil.* 84, 1185–1193. doi: 10.1016/s0003-9993(03)00030-3

Huang, V., and Krakauer, J. (2009). Robotic neurorehabilitation: a computational motor learning perspective. *J. Neuroeng. Rehabil.* 6:5. doi: 10.1186/1743-0003-6-5

Hussain, S., Xie, S. Q., and Jamwal, P. K. (2013a). Adaptive impedance control of a robotic orthosis for gait rehabilitation. *IEEE Trans. Syst. Man Cybernet. Part B* 43, 1025–1034. doi: 10.1109/tsmcb.2012.2222374

Hussain, S., Xie, S. Q., and Jamwal, P. K. (2013b). Robust nonlinear control of an intrinsically compliant robotic gait training orthosis. *IEEE Trans. Syst. Man Cybernet.* 43, 655–665. doi: 10.1109/tsmca.2012.2207111

Israel, J. F., Campbell, D. D., Kahn, J. H., and Hornby, T. G. (2006). Metabolic costs and muscle activity patterns during robotic- and therapist-assisted treadmill walking in individuals with incomplete spinal cord injury. *Phys. Ther.* 86, 1466–1478. doi: 10.2522/ptj.20050266

Jamwal, P. K., Xie, S. Q., Hussain, S., and Parsons, J. G. (2014). An adaptive wearable parallel robot for the treatment of ankle injuries. *IEEE/ASME Trans. Mechatr.* 19, 64–75. doi: 10.1109/tmech.2012.2219065

Kollen, B. J., Lennon, S., Lyons, B., Wheatley-Smith, L., Scheper, M., Buurke, J. H., et al. (2009). The effectiveness of the bobath concept in stroke rehabilitation what is the evidence? *Stroke* 40, E89–E97. doi: 10.1161/STROKEAHA.108.533828

Koopman, B., van Asseldonk, E. H. F., and van der Kooij, H. (2013). Selective control of gait subtasks in robotic gait training: foot clearance support in stroke survivors with a powered exoskeleton. *J. Neuroeng. Rehabil.* 10:3. doi: 10.1186/1743-0003-10-3

Kwakkel, G. (2009). Intensity of practice after stroke: more is better. *Schweizer Arch. Neurol. Psychiatr.* 160, 295–298.

Kwakkel, G., Kollen, B. J., and Krebs, H. I. (2008). Effects of robot-assisted therapy on upper limb recovery after stroke: a systematic review. *Neurorehabil. Neural Repair* 22, 111–121. doi: 10.1177/1545968307305457

Langzam, E., Isakov, E., and Mizrahi, J. (2006a). Evaluation of methods for extraction of the volitional EMG in dynamic hybrid muscle activation. *J. Neuroeng. Rehabil.* 3:27.

Langzam, E., Nemirovsky, Y., Isakov, E., and Mizrahi, J. (2006b). Muscle enhancement using closed-loop electrical stimulation: volitional versus induced torque. *J. Electromyogr. Kinesiol.* 17, 275–284. doi: 10.1016/j.jelekin.2006.03.001

Lee, S., Shin, S. Y., Lee, J. W., and Kim, C. (2013). Design of an 1 DOF assistive knee joint for a gait rehabilitation robot. *J. Korea Robot. Soc.* 8, 8–19. doi: 10.7746/jkros.2013.8.1.008

Lum, P. S., Burgar, C. G., and Shor, P. C. (2004). Evidence for improved muscle activation patterns after retraining of reaching movements with the MIME robotic system in subjects with post-stroke hemiparesis. *IEEE Trans. Neural. Syst. Rehabil. Eng.* 12, 186–194. doi: 10.1109/tnsre.2004.827225

Marchal-Crespo, L., and Reinkensmeyer, D. (2009). Review of control strategies for robotic movement training after neurologic injury. *J. Neuroeng. Rehabil.* 6:20. doi: 10.1186/1743-0003-6-20

Maffiuletti, N., Minetto, M., Farina, D., and Bottinelli, R. (2011). Electrical stimulation for neuromuscular testing and training: state-of-the art and unresolved issues. *Eur. J. Appl. Physiol.* 111, 2391–2397. doi: 10.1007/s00421-011-2133-7

Metral, S., and Cassar, G. (1981). Relationship between force and integrated EMG activity during voluntary isometric anisotonic contraction. *Eur. J. Appl. Physiol. Occup. Physiol.* 46, 185–198. doi: 10.1007/bf00428870

Negi, S., Kumar, Y., and Mishra, V. M. (2016). "Feature extraction and classification for EMG signals using linear discriminant analysis," in *Proceedings of the International Conference on Advances in Computing, Communication, & Automation (ICACCA)*, Bareilly.

Phinyomark, A., Hirunviriya, S., Limsakul, C., and Phukpattaranont, P. (2010). "Evaluation of EMG feature extraction for hand movement recognition based on euclidean distance and standard deviation," in *Proceedings of the International Conference on Electrical Engineering/Electronics Computer Telecommunications and Information Technology (ECTI-CON)*, Chiang Mai.

Pittaccio, S., and Viscuso, S. (2011). An EMG-controlled SMA device for the rehabilitation of the ankle joint in post-acute stroke. *J. Mater. Eng. Perform.* 20, 666–670. doi: 10.1007/s11665-010-9826-7

Poli, P., Morone, G., Rosati, G., and Masiero, S. (2013). Robotic technologies and rehabilitation: new tools for stroke patients' therapy. *Biomed Res. Int.* 2013:153872.

Popović, D. (2014). Advances in functional electrical stimulation (FES). *J. Electromyogr. Kinesiol.* 24, 795–802. doi: 10.1016/j.jelekin.2014.09.008

Roy, A., Krebs, H. I., Williams, D. J., Bever, C. T., Forrester, L. W., Macko, R. M., et al. (2009). Robotaided neurorehabilitation: a novel robot for ankle rehabilitation. *IEEE Trans. Robot.* 25, 569–582. doi: 10.1109/tro.2009.2019783

Saglia, J. A., Tsagarakis, N. G., Dai, J. S., and Caldwell, D. G. (2012). Control strategies for patientassisted training using the ankle rehabilitation robot (ARBOT). *IEEE/ASME Trans. Mechatr.* 99, 1–10.

Vallery, H., van Asseldonk, E. H. F., Buss, M., and van der Kooij, H. (2009). Reference trajectory generation for rehabilitation robots: complementary limb motion estimation. *IEEE Trans. Neural. Syst. Rehabil. Eng.* 17, 23–30. doi: 10.1109/TNSRE.2008.2008278

Veer, K., and Sharma, T. (2016). A novel feature extraction for robust EMG pattern recognition. *J. Med. Eng. Technol.* 40, 149–154. doi: 10.3109/03091902.2016.1153739

Veerbeek, J. M., Langbroek-Amersfoor, A. C., van Wegen, E. E., Meskers, C. G., and Kwakkel, G. (2017). Effects of robot-assisted therapy for the upper limb after stroke: a systematic review and meta-analysis. *Neurorehabil. Neural Repair* 31, 107–121. doi: 10.1177/1545968316666957

Veneman, J. F., Kruidhof, R., Hekman, E. E. G., Ekkelenkamp, R., Van Asseldonk, E. H. F., and van der Kooij, H. (2007). Design and evaluation of the LOPES exoskeleton robot for interactive gait rehabilitation. *IEEE Trans. Neural. Syst. Rehabil. Eng.* 15, 379–386. doi: 10.1109/tnsre.2007.903919

Wist, S., Clivaz, J., and Sattelmayer, M. (2016). Muscle strengthening for hemiparesis after stroke: a meta-analysis. *Ann. Phys. Rehabil. Med.* 59, 114–112. doi: 10.1016/j.rehab.2016.02.001

Wolbrecht, E. T., Chan, V., Reinkensmeyer, D. J., and Bobrow, J. E. (2008). Optimizing compliant, model-based robotic assistance to promote neurorehabilitation. *IEEE Trans. Neural. Syst. Rehabil. Eng.* 16, 286–297. doi: 10.1109/TNSRE.2008.918389

Wolf, S. L., Lecraw, D. E., and Barton, L. A. (1989). Comparison of motor copy and targeted biofeedback training techniques for restitution of upper extremity function among patients with neurologic disorders. *Phys. Ther.* 69, 719–735. doi: 10.1093/ptj/69.9.719

Generalize Robot Learning From Demonstration to Variant Scenarios With Evolutionary Policy Gradient

Junjie Cao[1], Weiwei Liu[1], Yong Liu[1]* and Jian Yang[2]

[1] Institute of Cyber Systems and Control, Zhejiang University, Hangzhou, China, [2] China Research and Development Academy of Machinery Equipment, Beijing, China

There has been substantial growth in research on the robot automation, which aims to make robots capable of directly interacting with the world or human. Robot learning for automation from human demonstration is central to such situation. However, the dependence of demonstration restricts robot to a fixed scenario, without the ability to explore in variant situations to accomplish the same task as in demonstration. Deep reinforcement learning methods may be a good method to make robot learning beyond human demonstration and fulfilling the task in unknown situations. The exploration is the core of such generalization to different environments. While the exploration in reinforcement learning may be ineffective and suffer from the problem of low sample efficiency. In this paper, we present Evolutionary Policy Gradient (EPG) to make robot learn from demonstration and perform goal oriented exploration efficiently. Through goal oriented exploration, our method can generalize robot learned skill to environments with different parameters. Our Evolutionary Policy Gradient combines parameter perturbation with policy gradient method in the framework of Evolutionary Algorithms (EAs) and can fuse the benefits of both, achieving effective and efficient exploration. With demonstration guiding the evolutionary process, robot can accelerate the goal oriented exploration to generalize its capability to variant scenarios. The experiments, carried out in robot control tasks in OpenAI Gym with dense and sparse rewards, show that our EPG is able to provide competitive performance over the original policy gradient methods and EAs. In the manipulator task, our robot can learn to open the door with vision in environments which are different from where the demonstrations are provided.

Keywords: learning from demonstration, generalization, exploration, reinforcement learning, evolutionary algorithms

1. INTRODUCTION

Hand-engineering of a controller is the basic approach to make robots autonomous for a certain task. For tasks whose execution depends on the circumstances of the environment or the interaction with human, robots must handle complex perception. However, the hand-engineering method for such task is tired, especially for vision-based tasks which are exceptionally difficult. In order to overcome those problems, a significant research area in contemporary robotics centers on approaches where the robot controller is learned rather than programmed (Rahmatizadeh et al., 2017). Learning from demonstration (LfD) allows humans to demonstrate a specific task to the

robot without having any knowledge about the robot's dynamic model or programming the control commands. One direct approach to learning from demonstration is Behavior Cloning, where human demonstrates the desired behavior to the robot – as supervisory signals of what the robot should do in the same states. However, the demonstrations are expensive to acquire and it is difficult to acquire complex manipulation skills just from demonstration.

On the other hand, the learned behavior is restricted to specific environment where human provide demonstrations. In order to generalize robot learning from demonstration to other scenarios, robot should to explore in the new situations. Reinforcement learning, learning to control through exploration by trial and error, provides a promising method for robot learning with generalization. For exploration, novel control policies are needed to gain diverse experience that is informative about the environment. The diverse experience is used to optimize the policies in return. In exploitation, the learning procedure exploits the good policy to collect state-action pairs with high rewards and further improve the performance of the good policy. In contrary to the exploration that may find best policy slowly and globally, exploitation aims to optimize the policy more efficiently and locally. To generalize robot learning and be sample efficient, preventing the mechanical wear, it usually needs to trade-off between exploration and exploitation in interaction with the real world. Proposing efficient and effective robot learning method has always been the hot spot in robotics.

Random perturbation of the agent's action is the classic method to induce novel behaviors, such as ϵ-greedy for Q-learning (Sutton and Barto, 1998) and policy gradient with action noise (Silver et al., 2014). However, action noise is usually independent of states and in fact is local perturbation, which generates unsmooth trajectories and is unlikely to produce various large-scale behavioral patterns for effective exploration (Osband et al., 2017). Recent works (Fortunato et al., 2017; Plappert et al., 2017; Gangwani and Peng, 2018) have show that exploration with parameter noise outperforms action noise, especially in tasks where the reward is sparse. Instead of promoting exploration with parameter perturbation, a lot of other explore strategies rely on the state novelty to increase the diversity of experience and obtain outstanding results in computer games (Bellemare et al., 2016; Houthooft et al., 2016; Pathak et al., 2017; Tang et al., 2017).

Evolutionary Algorithms (EAs) have been successfully used to optimize policy represented with neural network (Such et al., 2017) by perturbing and searching directly in the policy parameter space. The policy parameter evaluation of EAs is based on the cumulative reward received in the whole episode. Thus, EAs optimize the policy with more comprehensive insight and can solve the sequential decision making problems that have sparse reward signals.

Policy gradient methods are usually used to search best policy greedily, promoting better exploitation. As the original version of policy gradient algorithm, REINFORCE (Williams, 1992) tends to be of high variance due to the gradient estimation with Monte Carlo method. Actor-critic methods (Mnih et al., 2016)

use the value function to reduce the variance and improve the performance of policy gradient. In order to improve the accuracy of gradient estimation and stabilize the learning procedure, Schulman et al. (2015, 2017) constrained the step size of policy gradient descent within a local area of previous policy. Results from previous research show that policy gradient methods are sample-efficient by taking advantage of the temporal structure of the experience. However, those policy gradient methods with action noise tend to converge to the local optimum, especially when the exploration is insufficient.

In this paper, we introduce Evolutionary Policy Gradient (EPG), incorporating the policy gradient methods with Evolutionary Algorithms (EAs). In our framework of evolutionary algorithm, population-based approach generates different policies with parameter perturbation to improve exploration. With demonstration guiding the evolutionary procedure, our EPG guides the robot to behave similarly to the demonstrations, which prevent robot exploring in unpromising area. By selecting elites based on the fitness metric that evaluates the cumulative reward of an entire episode, EPG pushes the next generation of policy toward regions that lead to higher probability of task accomplishment in the current situation. Thus, our EPG can explore to generalize with diverse policies and also prevent the ineffective exploration with demonstration guiding.

2. RELATED WORK

In recent years, Learning from demonstration (LfD) has been successfully used in the field of robotics for applications in autonomous helicopter maneuvers (Abbeel et al., 2010), playing table tennis (Calinon et al., 2010), multi-task manipulation (Rahmatizadeh et al., 2017), and deformable object manipulation (Matas et al., 2018). A major challenge in LfD is to extend these demonstrations to unseen situations (Argall et al., 2009). To mitigate this problem, one obvious way is to acquire a large number of demonstrations covering as many situations as possible. With limited demonstration data, Sylvain et al. (2007) and Calinon et al. (2009) propose to hand-engineer task-specific features. Different from those methods we use smaller number of demonstrations, but change the learning model by exploring in new situations to generalize better. We resort to reinforcement learning and evolutionary algorithm to learn from demonstration and generalize to unseen situations. Rajeswaran et al. (2018) also combined reinforcement learning with demonstration to accomplish complex dexterous manipulation tasks in the same environment as where the demonstrations come from. While our goal is to generalize the learning behavior with the demonstration from one specific situation to behave well in other situations. Nair et al. (2017) avoided invalid exploration in reinforcement learning with demonstration. Our experiments also present such benefits of demonstration in comparison with pure reinforcement learning, even though the demonstrations in our experiments is coming from environment with different configurations. We have not find any previous works focus on generalizing the

skill learning from demonstration to other situations. But we think the reinforcement learning after imitation learning can accomplish such goal and we also compare our algorithm with such method.

Exploration during learning from demonstration is the core of our work. And the lack of effective exploration are also the major challenges in Reinforcement Learning, especially in the environments with long time horizon and sparse reward. Many explore strategies, which rely on the state novelty, have been proposed to improve the diversity of trajectories (Bellemare et al., 2016; Houthooft et al., 2016; Pathak et al., 2017; Tang et al., 2017). One common point of those works is the need of complex supplementary structures to estimate the novelty which will introduce some additional sensitive hyper parameters and are suited mainly for exploration in video games. However, our EPG, which explore with the framework of EAs, is more general with an easy modification to the original policy gradient method.

ES and PGPE which explore with parameter perturbation can be regarded as Evolutionary Algorithm and are scalable to be implemented in parallel. Salimans et al. (2017) have demonstrated that ES is suited for the problem with long time horizon and delayed reward and does not need the approximation for the value function. PGPE (Sehnke et al., 2010) performs gradient based search in parameter space with low variance and is similar to ES. Wang et al. (2017) improve PGPE with EM-based policy exploration and an adaptive mechanism. To further improve exploration in ES, especially on sparse or deceptive Reinforcement Learning problems, Conti et al. (2017) hybrid novelty search and quality diversity algorithms with ES. Unlike ES, NES, and PGPE which are gradient based methods, recently, Such et al. (2017) evolve the weights of a deep neural network with Genetic Algorithm to solve RL problems. By comparing DDPG with CMA-ES, de Broissia and Sigaud (2016) conclude that policy gradient methods are significantly more sample-efficient than ES. And it is the same for other EAs, without taking advantage of temporal structure in the trajectories. Our EPG incorporates the policy gradient into the framework of EAs to exploit the sample efficiency of policy gradient methods.

Recent works (Fortunato et al., 2017; Plappert et al., 2017) proposed to explore by adding noise to the parameter space and optimizing policy with gradient descent. Their results have shown that parameter perturbation can successfully be combined with reinforcement learning algorithms and often lead to improved performance compared to adding noise in action space. Similar to those works, our EPG also combines parameter perturbation with policy gradient methods. To further improve the exploration and parallelizability, our EPG resorts to the framework of EAs. Inspired by the Genetic Algorithms, Gangwani and Peng (2018) mutate the policy with policy gradient methods. However, without perturbing policy parameter vector during policy evolution, the diversity of policies and exploration in their method are limited. Our EPG retains the major benefits of the recent works, and the whole procedure can be approximated to the optimization of an objective function that evaluates the Gaussian distribution of policy parameter.

3. BACKGROUND

3.1. Learning From Demonstration (LfD) With Behavior Cloning

In recent years, LfD was successfully used in the field of robotics. Behavior cloning, as the simplest learning from demonstration method, can be performed using standard, efficient supervised learning methods. Compared with reinforcement learning methods that learn from scratch, behavior cloning requires fewer interactions with the environment.

Provided with the observation-action pairs, behavior cloning can fit a stochastic policy with supervise learning, mapping observations to distributions of action directly, just by maximizing the log likelihood of the demonstrated actions:

$$L = -E_{(s,a)\sim demo}[\log \pi(a|s)]. \tag{1}$$

where (s, a) represents the state-action pair from demonstrations, E represents expectation over (s, a) and π is the stochastic policy to be optimized.

3.2. Policy Gradient and Explore With Action Noise

Reinforcement Learning (RL) is popular in solving sequential decision making problems where a robot interacts with an environment, sequentially choosing an action a_t according to a policy $\pi(a|s)$ based on the state s_t at time t. After taking the action a_t, state s_t transforms to the next state s_{t+1}. And the robot receives a scalar reward $r(s_t, a_t)$ from the environment. In the Markovian environment, the probability distribution over the next state s_{t+1}, called transition probability, is satisfying Markov property, i.e., $s_{t+1} \sim p(s_{t+1}|s_t, a_t)$. The objective of robot learning with RL is to obtain a policy π which maximizes the expected discounted cumulative reward, i.e., $J(\pi) = E_\tau\left[\sum_t \gamma^t r(s_t, a_t)\right]$, where γ is the discounted factor that trade-off between shorter and longer term rewards. Solving such problem can be modeled as Markov Decision Process (MDP).

Policy Gradient method is one kind of reinforcement learning algorithms. For exploration in action space, stochastic policy samples from a Gaussian distribution $\pi_\theta \sim \mathcal{N}(\mu(s), \sigma(s)^2 I)$ with $\mu(s)$ and $\sigma(s)$ parameterized by θ, at each time step. Stochastic policy gradient methods maximize the expected cumulative reward by estimating the performance gradient $\nabla_\theta J(\pi_\theta)$ based on the Stochastic Policy Gradient Theorem (Sutton et al., 2000). For deterministic policy gradient methods, such as DPG (Silver et al., 2014) and DDPG (Lillicrap et al., 2015), the critic estimates the state-action value function $Q(s, a)$ using off-policy data which is sampled with a noisy policy. The noisy policy improves the exploration by adding additive action noise to deterministic policy: $\hat{\pi}_\theta(s) = \pi_\theta(s) + w$, where w represents the action noise with its variance annealing to trade-off between exploration and exploitation.

3.3. Policy Search With Evolutionary Algorithms

Most real-world problems can be modeled as MDP in which agents or robots only receive a reward signal after a series of

actions. In the MDP where rewards are sparse, it is difficult to associate actions with rewards. This situation is often denoted as the temporal credit assignment problem (Sutton and Barto, 1998).

Inspired by natural selection, Evolutionary Algorithms (EAs) are a series of black box optimization methods which are heuristic search procedures with several operators: new solution generation, mutation, selection, crossover and so on. Evolutionary Algorithms for sequential decision making problems are invariant to sparse rewards with long time horizons (Fortunato et al., 2017). Population-based approach in EAs has the advantage of promoting exploration, by parameter perturbation (mutation). The redundancy in a population and the selection of elites improve the robustness and stability of the heuristic search procedure. In computation complexity, EAs outperform back propagation methods in optimizing neural network with only forward evaluation of the parameter. Because of these merits of EAs, a number of recent research in RL problems have used EAs as an alternative to standard RL algorithms. Such et al. (2017) use genetic algorithms (GAs) to train deep neural networks for policy search. Conti et al. (2017) and Salimans et al. (2017) indicate that evolutionary strategies (ES) is scalable alternative to Reinforcement Learning and can improve exploration in RL.

In policy gradient methods, after sampling on each time step, the gradient is calculated by differentiating the policy with respect to the parameters. However, the derivative of the policy may not exist or be difficult to calculate. And sampling from the noisy policy on each time step leads to the noisy gradient estimation. Some EAs address such variance problem by replacing the random action sampling with parameter vector sampling, like natural evolution strategies (NES) (Wierstra et al., 2008) and policy gradients with parameter-based exploration (PGPE) (Sehnke et al., 2010). These algorithms represent the population with a probability distribution $p(\theta|\rho)$ over policy parameters θ, where ρ is the parameter of the distribution. Instead of sampling action at each time step with stochastic policy, PGPE samples a policy parameter vector from $p(\theta|\rho)$ to construct a deterministic policy, from which actions are taken. So PGPE addresses the variance in trajectory and noisy gradient problems by generating an entire trajectory with only one parameter vector sampled before exploration. The objective function to be maximized by searching ρ with stochastic gradient ascent is the expected cumulative reward over all parameter vectors:

$$J(\rho) = \int_\theta \int_\tau p(\tau, \theta|\rho) R(\tau) d\tau d\theta, \qquad (2)$$

where $R(\tau)$ represents the cumulative reward in a trajectory τ. Differentiating this objective function with respect to probability distribution parameter ρ, the gradient can be estimated by sampling θ from $p(\theta|\rho)$, then running the policy with parameter θ to generate trajectory τ, which submits $\tau \sim p(\tau|\theta)$. PGPE will choose Gaussian distribution as the policy parameter probability distribution, i.e., $p(\theta|\rho) = \mathcal{N}(\mu, \sigma^2 I)$ with $\rho = [\mu, \sigma]$. Optimizing μ and keeping σ as a constant, PGPE (Sehnke et al., 2010) reduces to evolution strategies (ES) (Salimans et al., 2017).

The ES, NES, and PGPE, introduced above, perform stochastic gradient descent with the calculation of gradient similar to the finite-difference methods, and are gradient-based algorithms. As another kind of classical Evolutionary Algorithms (EAs), a truly gradient-free method, Genetic Algorithm can also train deep neural networks for policy to solve the challenging sequential decision making problem. However, EAs for policy search do not exploit the information of each state-action pair in trajectories which make the policy gradient algorithms more sample-efficient. Thus, EAs need more samples of environment interaction (de Broissia and Sigaud, 2016).

4. METHOD: EVOLUTIONARY POLICY GRADIENT

Based on the framework of EAs, in this work, we define the policy with parameter vector θ and fixed neural network structure as individual. The mutation operator in our EPG includes random mutation and optimal mutation. EPG selects some elites, i.e., good policy parameter vectors according to the evaluation of the fitness function. As most EAs for reinforcement learning problems, the fitness function in EPG is defined as the average cumulative rewards of several episodes, where the rewards are received by robot after accomplishing tasks. Usually, EAs perform crossover directly in parameter space (Floreano et al., 2008) to increase the diversity of population. Instead, the crossover operator in our EPG combines the elites in the action distribution of the policies. And the combination is highly relied on the demonstration guided imitation learning.

Evolutionary algorithms (EAs), usually regarded as black box optimization processes, are heuristic and lack the theoretical guarantee. Although EPG is based on the framework of EAs, in this section, we will first derive the stochastic policy gradient method for the perturbed policies in mathematics and then propose our improvement to fit the framework of EAs with demonstrations guiding its procedure.

4.1. Optimization of Perturbed Policies With Policy Gradient

To achieve structured exploration, EPG will perturb current policy parameter vector to form a set of policies by applying additive Gaussian noise to the parameter vector of the current policy: $\theta = \theta' + \mathcal{N}(0, \sigma^2 I)$. The policy parameter perturbation is actually sampling parameters from the probability distribution $p(\theta|\rho)$ with $\rho = [\mu, \sigma]$, and μ is equal to the current policy parameter vector θ'. EPG optimizes the mean of the probability distribution of the parameters with gradient descent, while initiates the variance of the distribution and anneals it during the training procedure. The initial σ represents the capability of exploration and can be tuned according to the sparsity of rewards. The objective function here is the same as that of PGPE and ES (i.e., Equation 2).

Now, we derivate the gradient of the same objective $J(\rho)$ in a different way as in PGPE (Sehnke et al., 2010). After rewriting equation (2) in the discrete format, noting τ is conditionally

independent of ρ given θ, so $p(\tau, \theta | \rho) = p(\tau | \theta) p(\theta | \rho)$, we have:

$$J(\rho) = \sum_{\epsilon \sim \mathcal{N}(0, I)} \sum_{\tau} p(\tau | \epsilon \sigma + \mu) R(\tau). \tag{3}$$

By sampling $\epsilon \sim \mathcal{N}(0, I)$ to construct policy parameters and executing the policy to generate τ, the gradient of $J(\rho)$ with respect to probability distribution parameter ρ can be calculated:

$$\nabla_\rho J(\rho) = \frac{1}{N} \sum_{\substack{\epsilon \sim \mathcal{N}(0, I) \\ \tau \sim p(\tau | \epsilon \sigma + \mu)}} \nabla_\rho \log p(\tau | \epsilon \sigma + \mu) R(\tau). \tag{4}$$

For finite horizontal Markov decision process (MDP) with trajectory $\tau = [s_{1:T}, a_{1:T}]$, in which $s_{1:T}, a_{1:T}$ represents the sequence of state and action pairs, we have:

$$\log p(\tau | \theta) = \sum_{t=0}^{T} \log p(s_{t+1} | s_t; a_t) + \log \pi_\theta(a_t | s_t). \tag{5}$$

Substituting equation (5) into (4) and noting that we only optimize μ, we replace ρ with μ and denote $\pi_{\epsilon \sigma + \mu}$ with π_μ for clarity:

$$\nabla_\mu J(\mu) = \frac{1}{N} \sum_{\epsilon \sim \mathcal{N}(0, I)} \sum_{t=0}^{T} \nabla_\mu \log \pi_\mu(a_t | s_t) R(\tau). \tag{6}$$

To reduce the variance of the gradient estimation, the cumulative reward $R(\tau)$ can be substituted with the advantage function $A^{\pi_\mu}(s_t, a_t)$, which represents the improvement in cumulative reward obtained so far by taking action a_t in state s_t. According to the previous work (Mnih et al., 2016), we calculate the advantage function with an approximated value function and the obtained cumulative reward.

Proximal policy optimization algorithms (PPO) (Schulman et al., 2017) optimizes a "surrogate" objective function including a penalty term to constrain the size of the policy update. The updating equation for optimizing equation (2) with PPO can be derived as:

$$\nabla_\mu J(\mu) = \frac{1}{N} \sum_{\epsilon} \Big[\sum_t A^{\pi_{\mu_{old}}}(s_t, a_t) \frac{\nabla_\mu \pi_\mu(a_t | s_t)}{\pi_{\mu_{old}}(a_t | s_t)}$$
$$- \beta KL \big[\pi_{\mu_{old}}(\cdot | s_t), \pi_\mu(\cdot | s_t) \big] \Big], \tag{7}$$

where $\pi_\mu, \pi_{\mu_{old}}$ represent the current policy and the policy after previous iteration, respectively. $KL \big[\pi_{\mu_{old}}(\cdot | s_t), \pi_\mu(\cdot | s_t) \big]$ is the Kullback–Leibler divergence between the action distributions of the two policies, and β works as the coefficient of the penalty term.

4.2. The Framework of EPG

Our EPG algorithm is derived and approximated from the optimization of perturbed policies with policy gradient methods, by adding some heuristic of EAs. As EAs differ in how to perform those operators, we define the mutation, selection and crossover to form the framework of our EPG.

For exploration, EPG generates a population of N individuals (policy parameter vector θ_i) by parameter perturbation, which applies additive Gaussian noise to the parameter vector of the current policy: $\theta_i = \mu + \sigma \epsilon$, where $\epsilon \sim \mathcal{N}(0, I)$. The parameter perturbation in EPG can be regarded as the random mutation of the individual. After executing each perturbed policy (individual) for several episodes, the fitness can be evaluated by averaging the cumulative rewards received during those episodes. Then EPG performs truncation selection, where the top n individuals become the elites. For each elite individual, EPG optimizes the policies with policy gradient algorithms, which can be regarded as optimal mutation. Those policy gradient optimization processes can be implemented in parallel to accelerate the training procedure. After the optimal mutation, it is easy to combine those elite optimal policies to one policy by averaging their parameter vectors.

Using $\nabla_\mu J(\mu)$, Equation (7), to update the mean of policy parameter vector μ, noting $\theta_i = \epsilon_i \sigma + \mu$ and $\epsilon_i \sim \mathcal{N}(0, I)$, we get:

$$\hat{\theta} = \mu + \frac{1}{N} \sum_{i=1}^{N} \sum_t A^{\pi_{\theta_i}}(s_t, a_t) \nabla_\mu \log \pi_{\epsilon_i \sigma + \mu}(a_t | s_t)$$
$$\approx \frac{1}{n} \sum_{i=1}^{n} \Big[\mu + \epsilon_i \sigma + \sum_t A^{\pi_{\theta_i}}(s_t, a_t) \nabla_\mu \log \pi_{\epsilon_i \sigma + \mu}(a_t | s_t) \Big]$$
$$= \frac{1}{n} \sum_{i=1}^{n} [\theta_i + \Delta \theta_i]. \tag{8}$$

where $\Delta \theta_i$ is calculated with policy gradient methods with regard to policy parameter vector θ_i. Equation (8) shows that the framework mentioned above is actually some kind of approximation to the gradient descent procedure of optimizing objective function (2).

The selection of n elites out of the population by choosing n best policies is a technique called elitism in Evolutionary Algorithms (EAs). The mutation in EPG includes random mutation by policy parameter perturbation and optimal mutation by policy optimization with policy gradient. The crossover in the parameter space gives few reasonable explanation. We resort to the ensemble learning to crossover those elite policies by learning a classifier to choose which elite policy to take action at every time step. Combining the classifier and the optimized elite polices, we get a compound policy. In order to generate the child policy for the next generation, we first initialize the child policy with the average parameter vector of elite polices, then minimize the KL divergence between the action distributions of the child policy and the compound policy. The overall crossover operator in our EPG can be regarded as the crossover in the action space, which seems to be more reasonable since the output of policy is action. The crossover operator is also equivalent to imitation learning after ensemble, combining the diverse elite policies. The objective of imitation learning can be augmented with that of behavior clone to guide the evolutionary process with demonstration. After the crossover, we get one individual (child policy) on which random mutation (parameter

FIGURE 1 | Framework of EPG.
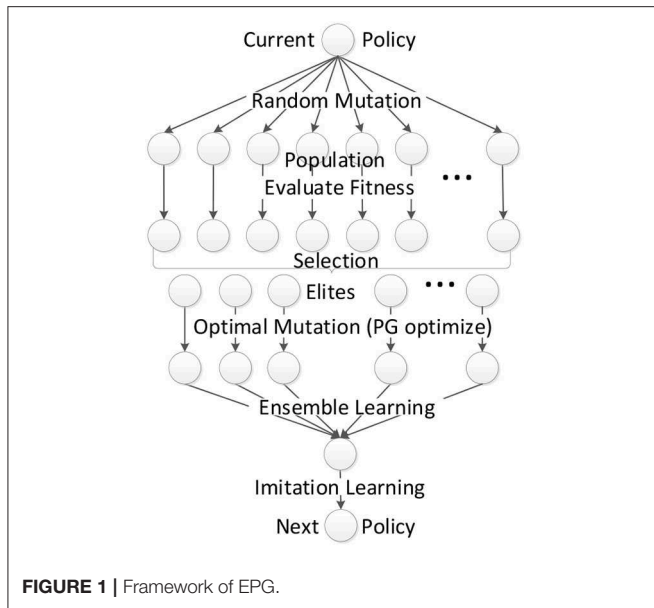
perturbation) can be implemented. It means a new generation (iteration) begins.

The whole EPG algorithm is scalable to be implemented in parallel in different computer nodes with different random seeds. Algorithm 1 illustrate the procedure in one single node of our EPG. **Figure 1** illustrates the complete learning procedure of EPG. In the rest of this section, the mutation and crossover operators will be detailed.

Algorithm 1: The procedure in node i of EPG.

1: **repeat**
2:     Random mutate (perturb) $\pi_\theta \rightarrow \pi_{\theta_i}$;
3:     Execute policy π_{θ_i} for several episodes to collect rollout trajectories τ;
4:     Optimize policy with policy gradient: $\pi_{\theta_i} \rightarrow \hat{\pi}_{\theta_i}$;
5:     Send the parameters of $\hat{\pi}_{\theta_i}$, cumulative rewards R_i and visited states S_i in τ to every other nodes;
6:     Calculate $fitness(\pi_{\theta_j})$ by averaging R_j in every nodes;
7:     Select optimized elite policies $\hat{\pi}_{\theta_j}$ according to $fitness(\pi_{\theta_j}), j \in [1, n]$;
8:     Child policy $\pi'_\theta \leftarrow Crossover(\hat{\pi}_{\theta_{1:n}}, S_{1:n}, R_{1:n}, demo)$;
9: **until** k times of evolution loop.

4.3. Mutation Operator in EPG

Similar to most EAs in solving sequential decision making problems, EPG operates mutation by perturbing current policy to generate a population of policies. Parameter perturbation, called random mutation, is implemented on stochastic policy instead of deterministic policy as policy gradients with parameter-based exploration (PGPE) (Sehnke et al., 2010). The stochastic policy outputs the mean and variance of a Gaussian distribution from which action is sampled. The output variance of stochastic policy and the variance of parameter perturbation determine

the exploration in action and parameter space, respectively. By combining the exploration both in action space and parameter space, EPG can avoid local minima more easily.

After selecting elites from the population, EPG modifies each elite with policy gradient methods in parallel, which is regarded as optimal mutation. The policy gradient is estimated with the rollout samples collected by stochastic policy in environment with different random seeds, thus induces randomness for elite policies update. So the optimal mutation operator still maintains sufficient diversity of the population and good exploration in the state space. The optimal mutation improves EPG's efficiency in the usage of sampling data, by taking advantage of the powerful gradient descent method and reusing the rollout samples generated for elite selection.

Algorithm 2: Crossover operator in node i of EPG.

Require: $\hat{\pi}_{\theta_j}, S_j, R_j, j \in [1, n]$ of elite policies; Demonstrations from one fixed environment: $(s, a) \sim demo$
Ensure: Child policy π'_θ, generalized to a different situation as in demonstrations;
1: For s in the k'th trajectory of elite j: $\omega_s = \frac{R_{jk} - \min R_j}{\max R_j - \min R_j}$;
2: Send the gradient of (9): $\nabla_c J_j^{ML}$ to other nodes;
3: Train $\pi_c(\hat{\pi}_{\theta_j}|s)$ with gradient: $\frac{1}{n} \sum_j \nabla_c J_j^{ML}$;
4: Combine classifier and elite policies to form $\pi_{exp}(\cdot|s)$
5: Average $\theta = \frac{1}{n} \sum_{j=1}^n \theta_j$ to initialize child policy π'_θ;
6: Initialize data set: $\{\hat{\tau}_i\}$;
7: **repeat**
8:     Generate one trajectory τ with π'_θ;
9:     Aggregate data set $\{\hat{\tau}_i\}$ with τ;
10:     Calculate the gradient of (10): $\nabla_\theta J_j^{IL}$ with $\{\hat{\tau}_i\}$;
11:     Send $\nabla_\theta J_j^{IL}$ to other nodes;
12:     Update π'_θ with gradient: $\frac{1}{n} \sum_j \nabla_\theta J_j^{IL}$;
13: **until** t times of imitate iterations.

4.4. Crossover Operator in EPG

The crossover operator mixes parent policies and produces a new child policy. Those policies have identical network architectures. Floreano et al. (2008) produced the child policy by averaging the parameters of the parent policies, which called average crossover. Such crossover operation in parameter space dose not has steady performance, sometimes producing a worse policy than the policies before averaging. In our EPG framework, the finite number of population N usually results in the underestimation of policy gradient (6). It is more intuitive to crossover the policy in the distribution of action, since the output of policy is a distribution of action dependent on the state.

In our framework of EPG, every elite policies are trained independently with different random seeds, thus have distinct state distributions encountered. So we propose to first combine elite policies with ensemble learning, by learning a classifier $\pi_c(\hat{\pi}_{\theta_j}|s)$ to choose one elite policy $\hat{\pi}_{\theta_j}$ according to their visited state s. The intuition is that given a state we choose the elite policy, which encountered this state more often before,

to take action. The objective of the classifier is weighted maximum likelihood:

$$J^{ML} = - \sum_s \left[\omega_s \log \sum_j \pi_c(\hat{\pi}_{\theta_j}|s) \mathbb{1}_{s \in \tau_j} \right], \quad (9)$$

where $\mathbb{1}_{s \in \tau_j}$ is the indicator function and ω_s is the weight for state s, been the same in one trajectory. The formulation of ω_s is detailed in Algorithm 2, where R_j represents the set of cumulative rewards of several trajectories generated with elite j and R_{jk} is the k'th element of R_j. With the classifier and the elite policies, we get a compound policy: first select an elite policy and then take action according to it.

To inherit the essence of the optimized elite policies, our EPG crossovers them by imitating the compound policy, also called expert, with imitation learning. The expert, represented as $\pi_{exp} = \pi_c(\hat{\pi}_{\theta_j}(\cdot|s)|s)$, includes a classifier and several elite policies. Imitation learning is a method that develops new policy by mimicking expert's behaviors. The imitation is implemented in the action distribution by minimizing the KL divergence between the action distributions of the child policy and the compound policy: $KL[\pi_\theta, \pi_{exp}]$. Then, the resulting π_θ is an approximate Gaussian I-projection of π_{exp} which works as a good guiding distribution. To improve the sample efficiency, we initialize the child policy with the same architecture as elite policies and the average elite policies' parameter vectors. Moreover, in order to direct the policy evolution to more likely accomplish the task, especially at the beginning of exploration, we augment the objective of KL divergence with the negative log likelihood of demonstrated actions (Equation 1), resulting the objective function to be minimized:

$$J^{IL}(\theta) = \sum_s KL \left[\pi_\theta(\cdot|s), \pi_{exp}(\cdot|s) \right] - \lambda E_{(s,a) \sim demo}[\log \pi_\theta(a|s)]. \quad (10)$$

The λ in Equation (10) should decay to zero, promoting generalization to new environments. If the new environment is more different from where the demonstration is provided, smaller initial λ or larger decay factor should be chosen. To refine the result policy and avoid compounding errors due to the visited state distribution mismatch between the compound policy and the child policy, we run the result child policy for one episode to collect new samples to aggregate training set. With that training set, we optimize Equation (10) with the compound policy as supervisor. Then, with new child policy, the data collection and optimization processes can be executed again. This procedure can be iterated for several times and generates a final child policy that performs well under its own induced state distribution. And it is similar to the imitation learning with Dataset Aggregation (DAgger) algorithm (Ross et al., 2011). The KL-divergence in Equation (10) promotes high entropy in result policy, and thus encourages the exploration too. For Gaussian distribution, the surrogate loss (Equation 10) is easily optimized with stochastic gradient descent. The crossover operator in EPG is very sample-efficient. In experiments, we only iterated the Dataset Aggregation procedure for two or three times. The whole procedure of crossover operator is shown in Algorithm 2.

5. EXPERIMENTS

In the first part of this experiment section, we present the improved exploration capability of our method, comparing with the state-of-the-art reinforcement learning methods, where no demonstrations are provided. In the second part, we demonstrate the generalization capability of our method to different situations in robot learning from demonstration, where robot accommodates to new environments with active exploration. And we present that our method has better performance in generalization, in comparing with other related methods which combining imitation learning and reinforcement learning.

5.1. Exploration: Robot Control in State Space

Reinforcement Leaning methods are well-known as its capability of exploration in unknown environments without any instruction or demonstration from "expert." To present the improved exploration of our EPG, we set $\lambda = 0$ in Equation (10) and compare our EPG-PPO with the state-of-the-art reinforcement learning methods, ES and DPPO, in continuous control problems of OpenAI Gym (Brockman et al., 2016) without demonstrations. For comparison with related works and future research, we choose the average cumulative rewards, provided by the simulator in OpenAI Gym, as the evaluation criterion.

In our experiments we implement our EPG with PPO (Schulman et al., 2017) as policy gradient method, called EPG-PPO. DPPO is the abbreviation of "distributed proximal policy optimization" method (Heess et al., 2017) which is popular with its sample efficiency. In DPPO the distributed agents explore in environments with different random seeds and calculate the policy gradients which are averaged to update the policy. Our implementation of DPPO has the same full connected neural network structure as that of our EPG-PPO, composed with two hidden layers and "tanh" activation function. The dimension of each layer is ten times of the dimension of each robot's action space. And other major hyperparameters are similar, e.g., 2–5 nodes in parallel and learning rates within [0.0005, 0.001]. ES is the abbreviation of "evolutionary strategy" (Salimans et al., 2017) which is good at random exploration. In our implementation the policy is perturbed to form 5-8 individuals in population to generate experience and is updated according to the evaluation of those experience.

In some standard OpenAI Gym environments, such as "Reacher," "Hopper," "HalfCheetah," and "Swimmer," robots will definitely receive a task-related reward signal at each state with its value dependent on the state and action. **Figure 2** illustrate the scenes of robot control tasks in our experiments. In the first picture, the goal of "Reacher" is to control a two-arm robot with torque so as the end-effect approach the target point as near as possible, in 50 time steps. In the other three pictures, "Hopper," "HalfCheetah," and "Swimmer" are multi-joint robots which get more rewards by running or swimming forward away. **Figure 3** depicts the comparative performance in environments with dense rewards. Especially in "Swimmer," our EPG avoids the local minima which traps the PPO.
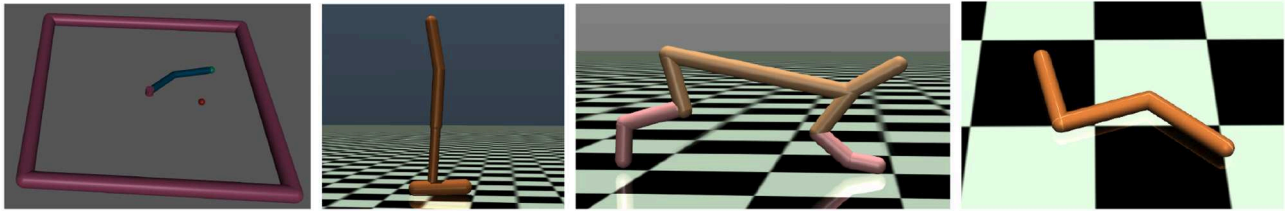
FIGURE 2 | Environments for the Robot Control in State Space: "Reacher," "Hopper," "HalfCheetah," and "Swimmer".
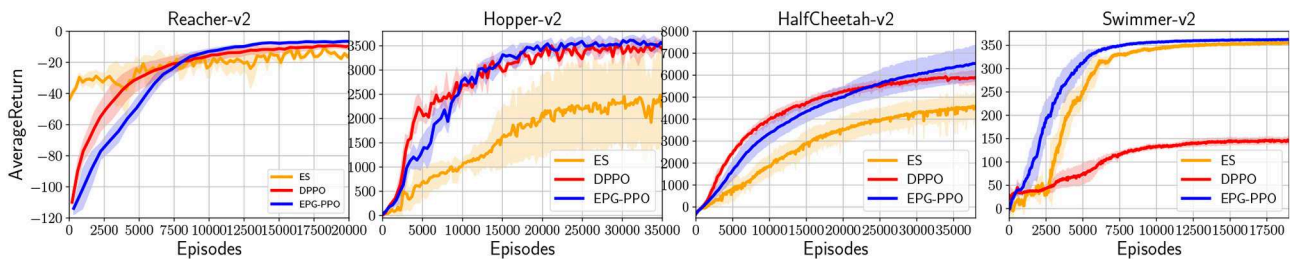
FIGURE 3 | Depict the average cumulative rewards of the policy after the crossover in EPG-PPO, the distributed policies in DPPO and the updated policy in ES, plotted over the episodes required during training. The solid lines represent the average performance in six repeated experiments with different random seeds and the shades exhibit the standard deviation of the performance in those repeated experiments.
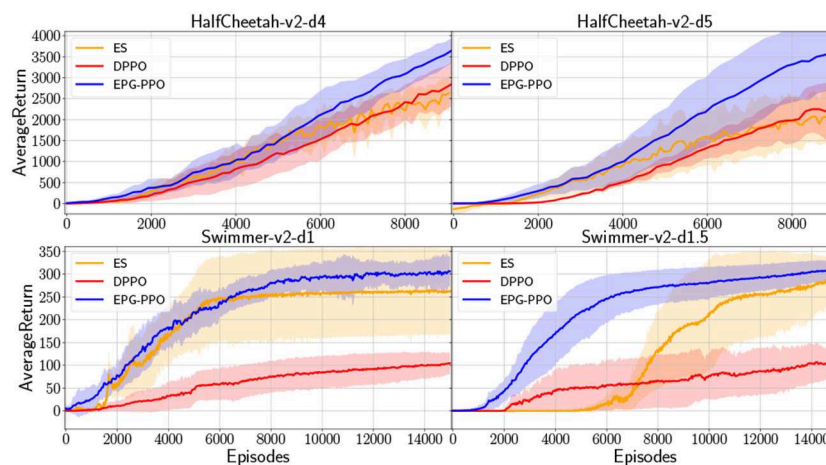
FIGURE 4 | In environments with sparse rewards, "HalfCheetah-v2-d4" and "HalfCheetah-v2-d5" provide reward signal until HalfCheetah's position in *x* axis surpass 4 and 5 units, respectively. "Swimmer-v2-d1" and "Swimmer-v2-d1.5" are modified with rewards delayed for 1 and 1.5 units in *x*, respectively.

In many real-world problems, robots may only receive rewards after the task accomplishment, where exploration play a more important role in robot learning. To construct environment with sparse reward, we modify the original gym environments, "HalfCheetah-v2" and "Swimmer-v2," to be with no reward at the beginning until the robot walk forward to surpass a threshold. In such situation robots must explore more to acquire reward signal and learn to accomplish the goal which, here in our experiments, is running forward as far as possible. The comparative results are illustrated in **Figure 4**, which depicts the performance of EPG-PPO, DPPO and ES, with the solid lines and shades representing the mean and standard deviation in six repeated experiments. In those environments with sparse rewards, the agents are initialized with random orientations and acquire performance with high

variance in repeated experiments. Although three algorithms all have worse performance than that in **Figure 3** due to the sparse reward, our EPG has more advantage with regard to the sample efficiency.

5.2. Generalization: Learning Vision Based Manipulation From Demonstration

To present the generalization of our EPG, in the complex manipulation task, we use simulation environment "Image48SawyerDoorPullHookEnv-v0" provided in "multiworld"[1]. In this task, the demonstrations, including images and corresponding displacements of the end effector, are

---

[1]https://github.com/vitchyr/multiworld

collected in original environment, while robot is learning and tested in different environments with distinct configurations of the door. The goal of this task is to open the door in 100 time steps. We train a neural network policy, which is composed with two convolutional layers and two full connected layers with 100 units for each. And we use "tanh" as the activation functions in hidden layers. The policy takes as input the images from the fixed camera and outputs the next displacement of the end-effector. The image viewed by the robot is with 48×48 pixels, shown in **Figure 5**. The robot receives a binary reward when the angle of door is above a threshold.

In the initial environment, we hand-engineer the robot to open the door and collect only one trajectory including images and the corresponding actions. We augment the demonstration dataset by adding minor noises to the actions with the images copied. Then, we change the configuration of the door, such as the position, initial angle and its handle's position. In the
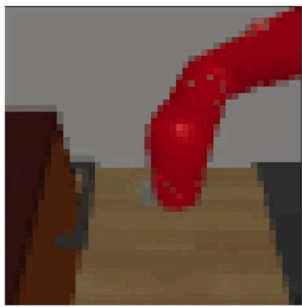
FIGURE 5 | Robot view.

new situation and with our EPG-PPO, robot interacts with the environment about thousands of time steps and learn from demonstration at the same time, robot can open the door with high success rate, as illustrated in **Figure 6**. While the robot learned with behavior clone without exploration failed in adapting to the new situation as depicted in **Figure 7**.

Even though we do not find any previous works aiming at generalizing skill learning from demonstration to different scenarios where the demonstrations are provided, the reinforcement learning after imitation learning is a good thought to accomplish such goal. To present the better generalization capability of our method, we implement "BC + DPPO" as baseline, where the robot is first trained with behavior clone to learn from demonstration, then trained with DPPO to generalize by exploration in new environments.

Figure 8 shows our experiments, where we change the positions of the door and its handle with an offset relative to their origins. In those experiments, we compare our EPG with "BC + DPPO." It is obvious that with the same few demonstrations, our EPG can learn from demonstration and generalize better. Even though the robot learns from the binary reward signal it receives, we evaluate our method by calculating the success rate of the task accomplishment, i.e., door is open, in 100 independent tests during the process of training. **Figure 8** shows the mean values and standard deviations of the success rates in five repeated experiments.

6. CONCLUSION AND DISCUSSION

In this paper we introduced a new learning from demonstration method, Evolutionary Policy Gradient (EPG), with
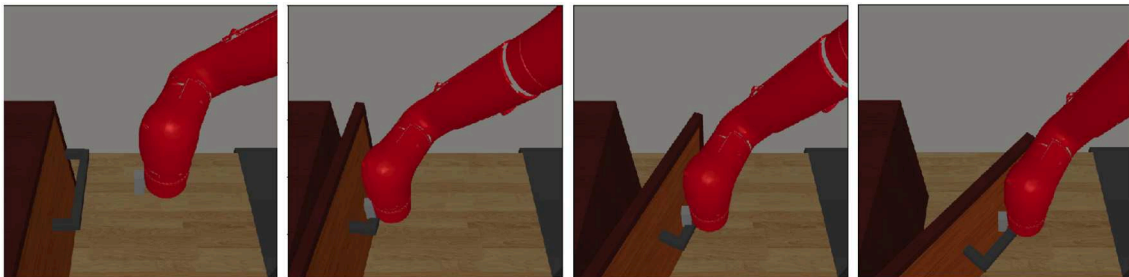
FIGURE 6 | Neural network policy trained with our EPG accomplish the task.

FIGURE 7 | Neural network policy trained with behavior clone failed.
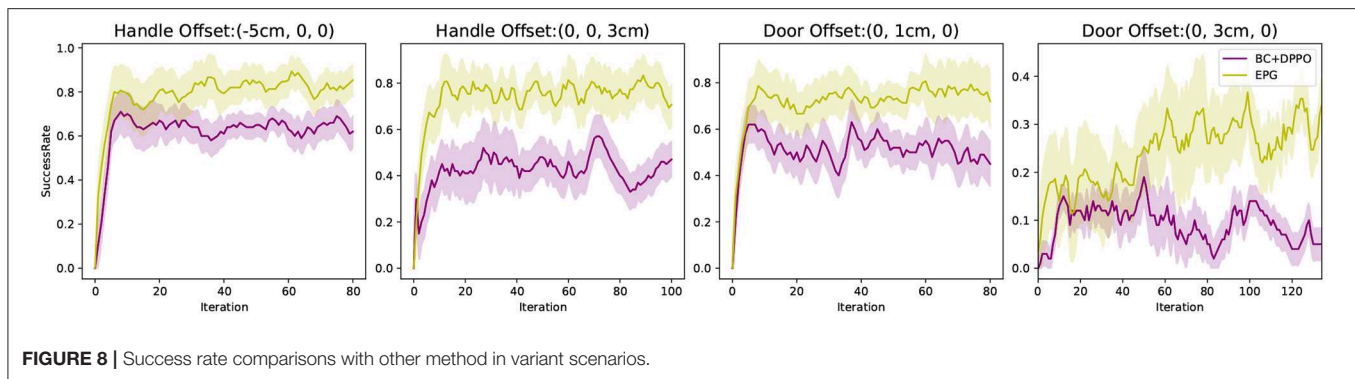
FIGURE 8 | Success rate comparisons with other method in variant scenarios.

demonstration guiding the evolution of policy. Just providing the demonstrations from one specific situation, our EPG can generalize the robot learning from the demonstrations to accomplish the same tasks in different environments.

With parameter perturbation and evolutionary framework, our EPG explore in the new environment to accomplish the task. With demonstration guiding and policy gradient optimization, robot can acquire the skill to accomplish the task with fewer interaction with the new environment. In the framework of EAs, EPG is scalable in parallel to accelerate the training process. Moreover, our EPG is a general framework and can be implemented with all kinds of policy gradient methods. The whole optimization procedure of EPG is based on the stochastic policy gradient theorem and behavior clone, with a little approximation to fit the framework of EAs.

Our aim is to improve the generalization of robot learning skill from demonstration. As the main contribution of our work, we present that the active exploration of robot can accomplish the goal of generalization. Next, we discuss the rationality behind our study from the perspectives of exploration, exploitation and generalization.

6.1. About Exploration
As a well-known Evolutionary Algorithm, evolutionary strategy (ES) only relies on exploring with parameter perturbation. When the reinforcement learning problem is with dense reward, our results shown in **Figure 3** demonstrated that ES has few advantage over DPPO which exploit the powerful policy gradient methods. On the other hand, ES outperforms DPPO in environment with sparse reward. Our results shown in **Figure 4** illustrate that parameter perturbation has more advantage in exploration. And, our EPG-PPO explores with parameter perturbation and retains the data efficiency of policy gradient. Thus, our method combines the merits of both ES and DPPO.

6.2. About Exploitation and Exploration Trade-Off
Our policy parameter perturbation with Gaussian noise is also a sample from Gaussian distribution with the optimized mean and annealing variance. The policy parameter sampling after every optimization step can be seen as a posterior sampling.

From this point of view, our EPG is an approximation to Thompson Sampling (Thompson, 1933) in the policy parameters. Thompson Sampling, originated from bandits problems, provides an elegant approach that tackles the exploration-exploitation dilemma. Previous works, inspired by Thompson Sampling, focus on problem with discrete action space by randomly selecting an action according to the probability it is optimal. Our method are aiming at problems with continuous action space. Improving our method toward the Exploration and Exploitation trade-off is a promising direction for future research. For example, we can take into the parameter variance optimization into consideration, making the parameter perturbation procedure bears more similarity to Thompson Sampling.

6.3. About Generalization
In the unknown environment without demonstration, robot trained with many state-of-the-art reinforcement learning methods, including DPPO, can learn to accomplish many complex tasks by interacting with the environment millions of time steps. In our experiments, we find that DPPO can not make robot explore to grasp the skill of opening the door in thousands of interaction time steps, due to the sparse reward. Exploration from scratch is sample inefficient and can not be seen as generalization, though with adaption to new environments. Our method learning from demonstration and adapting to new environments with few interactions are actually generalizing the learning skill from previous situation to new situation. Our deep combination of learning from demonstration and reinforcement learning presents a promising direction to improve the generalization of learning from demonstration and is worth further research. On the perspective of sample complexity, it is the generalization from demonstration that makes the reinforcement learning process more sample efficient. Thus, our method can also provide a new method to improve the sample efficiency of Reinforcement Learning methods.

DATA AVAILABILITY STATEMENT

Publicly available datasets were analyzed in this study. This data can be found here: https://github.com/vitchyr/multiworld, https://github.com/openai/gym.

AUTHOR CONTRIBUTIONS

FUNDING

REFERENCES

Abbeel, P., Coates, A., and Ng, A. Y. (2010). Autonomous helicopter aerobatics through apprenticeship learning. *Int. J. Robot. Res.* 29, 1608–1639. doi: 10.1177/0278364910371999

Argall, B. D., Chernova, S., Veloso, M., and Browning, B. (2009). A survey of robot learning from demonstration. *Robot. Auton. Syst.* 57, 469–483. doi: 10.1016/j.robot.2008.10.024

Bellemare, M., Srinivasan, S., Ostrovski, G., Schaul, T., Saxton, D., and Munos, R. (2016). "Unifying count-based exploration and intrinsic motivation," in *Advances in Neural Information Processing Systems* (Barcelona), 1471–1479.

Brockman, G., Cheung, V., Pettersson, L., Schneider, J., Schulman, J., Tang, J., et al. (2016). Openai gym. *arXiv* 1606.01540.

Calinon, S., D'Halluin, F., Caldwell, D. G., and Billard, A. G. (2009). "Handling of multiple constraints and motion alternatives in a robot programming by demonstration framework," in *IEEE-RAS International Conference on Humanoid Robots* (Paris). doi: 10.1109/ICHR.2009.5379592

Calinon, S., D'Halluin, F., Sauser, E. L., Caldwell, D. G., and Billard, A. G. (2010). Learning and reproduction of gestures by imitation. *Robot. Autom. Mag. IEEE* 17, 44–54. doi: 10.1109/MRA.2010.936947

Conti, E., Madhavan, V., Such, F. P., Lehman, J., Stanley, K. O., and Clune, J. (2017). Improving exploration in evolution strategies for deep reinforcement learning via a population of novelty-seeking agents. *arXiv* 1712.06560.

de Broissia, A. D. F., and Sigaud, O. (2016). Actor-critic versus direct policy search: a comparison based on sample complexity. *arXiv* 1606.09152.

Floreano, D., Dürr, P., and Mattiussi, C. (2008). Neuroevolution: from architectures to learning. *Evol. Intell.* 1, 47–62. doi: 10.1007/s12065-007-0002-4

Fortunato, M., Azar, M. G., Piot, B., Menick, J., Osband, I., Graves, A., et al. (2017). Noisy networks for exploration. *arXiv* 1706.10295.

Gangwani, T., and Peng, J. (2018). *Policy optimization by genetic distillation.*

Heess, N., Sriram, S., Lemmon, J., Merel, J., Wayne, G., Tassa, Y., et al. (2017). Emergence of locomotion behaviours in rich environments. *arXiv preprint arXiv:1707.02286.*

Houthooft, R., Chen, X., Duan, Y., Schulman, J., De Turck, F., and Abbeel, P. (2016). "Vime: variational information maximizing exploration," in *Advances in Neural Information Processing Systems* (Barcelona), 1109–1117.

Lillicrap, T. P., Hunt, J. J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., et al. (2015). Continuous control with deep reinforcement learning. *arXiv* 1509.02971.

Matas, J., James, S., and Andrew, J. D. (2018). Simto-real reinforcement learning for deformable object manipulation. *arXiv [preprint].* arXiv:1806.07851.

Mnih, V., Badia, A. P., Mirza, M., Graves, A., Lillicrap, T., Harley, T., et al. (2016). "Asynchronous methods for deep reinforcement learning," in *International Conference on Machine Learning* (New York, NY), 1928–1937.

Nair, A., Mcgrew, B., Andrychowicz, M., Zaremba, W., and Abbeel, P. (2017). Overcoming exploration in reinforcement learning with demonstrations. *arXiv* 1709.10089. doi: 10.1109/ICRA.2018.8463162

Osband, I., Russo, D., Wen, Z., and Van Roy, B. (2017). Deep exploration via randomized value functions. *arXiv* 1703.07608.

Pathak, D., Agrawal, P., Efros, A. A., and Darrell, T. (2017). "Curiosity-driven exploration by self-supervised prediction," in *International Conference on Machine Learning (ICML)* (Sydney, NSW). doi: 10.1109/CVPRW.2017.70

Plappert, M., Houthooft, R., Dhariwal, P., Sidor, S., Chen, R. Y., Chen, X., et al. (2017). Parameter space noise for exploration. *arXiv* 1706.01905.

Rahmatizadeh, R., Abolghasemi, P., Bölöni, L., and Levine, S. (2017). Vision-based multi-task manipulation for inexpensive robots using end-to-end learning from demonstration. *arXiv* 1707.02920. doi: 10.1109/ICRA.2018.8461076

Rajeswaran, A., Kumar, V., Gupta, A., Schulman, J., and Levine, S. (2018). Learning complex dexterous manipulation with deep reinforcement learning and demonstrations. *arXiv* 1709.10087. doi: 10.15607/RSS.2018.XIV.049

Ross, S., Gordon, G., and Bagnell, D. (2011). "A reduction of imitation learning and structured prediction to no-regret online learning," in *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics,* 627–635.

Salimans, T., Ho, J., Chen, X., Sidor, S., and Sutskever, I. (2017). Evolution strategies as a scalable alternative to reinforcement learning. *arXiv* 1703.03864.

Schulman, J., Levine, S., Abbeel, P., Jordan, M., and Moritz, P. (2015). "Trust region policy optimization," in *International Conference on Machine Learning,* 1889–1897.

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O. (2017). Proximal policy optimization algorithms. *arXiv* 1707.06347.

Sehnke, F., Osendorfer, C., Rückstieß, T., Graves, A., Peters, J., and Schmidhuber, J. (2010). Parameter-exploring policy gradients. *Neural Netw.* 23, 551–559. doi: 10.1016/j.neunet.2009.12.004

Silver, D., Lever, G., Heess, N., Degris, T., Wierstra, D., and Riedmiller, M. (2014). "Deterministic policy gradient algorithms," in *Proceedings of the 31st International Conference on Machine Learning* (Beijing).

Such, F. P., Madhavan, V., Conti, E., Lehman, J., Stanley, K. O., and Clune, J. (2017). Deep neuroevolution: genetic algorithms are a competitive alternative for training deep neural networks for reinforcement learning. *arXiv* 1712.06567.

Sutton, R. S., and Barto, A. G. (1998). *Reinforcement Learning: An Introduction.* Cambridge: MIT Press.

Sutton, R. S., McAllester, D. A., Singh, S. P., and Mansour, Y. (2000). "Policy gradient methods for reinforcement learning with function approximation," in *Advances in Neural Information Processing Systems* (Breckenridge, CO), 1057–1063.

Sylvain, C., Florent, G., and Aude, B. (2007). On learning, representing, and generalizing a task in a humanoid robot. *IEEE Trans. Syst. Man Cybern. B* 37, 286–298. doi: 10.1109/TSMCB.2006.886952

Tang, H., Houthooft, R., Foote, D., Stooke, A., Chen, O. X., Duan, Y., et al. (2017). "Exploration: a study of count-based exploration for deep reinforcement learning," in *Advances in Neural Information Processing Systems,* 2753–2762.

Thompson, W. R. (1933). On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika* 25, 285–294. doi: 10.1093/biomet/25.3-4.285

Wang, J., Eiji, U., and Kenji, D. (2017). Adaptive baseline enhances em-based policy search: validation in a view-based positioning task of a smartphone balancer. *Front. Neurorobot.* 11:1. doi: 10.3389/fnbot.2017.00001

Wierstra, D., Schaul, T., Peters, J., and Schmidhuber, J. (2008). "Natural evolution strategies," in *CEC 2008 IEEE Congress on Evolutionary Computation, 2008* (Hong Kong: IEEE), 3381–3387. doi: 10.1109/CEC.2008.4631255

Williams, R. J. (1992). Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Mach. Learn.* 8, 229–256. doi: 10.1007/BF00992696

Gaze Control of a Robotic Head for Realistic Interaction With Humans

Jaime Duque-Domingo*, Jaime Gómez-García-Bermejo and Eduardo Zalama

ITAP-DISA, University of Valladolid, Valladolid, Spain

When there is an interaction between a robot and a person, gaze control is very important for face-to-face communication. However, when a robot interacts with several people, neurorobotics plays an important role to determine the person to look at and those to pay attention to among the others. There are several factors which can influence the decision: who is speaking, who he/she is speaking to, where people are looking, if the user wants to attract attention, etc. This article presents a novel method to decide who to pay attention to when a robot interacts with several people. The proposed method is based on a competitive network that receives different stimuli (look, speak, pose, hoard conversation, habituation, etc.) that compete with each other to decide who to pay attention to. The dynamic nature of this neural network allows a smooth transition in the focus of attention to a significant change in stimuli. A conversation is created between different participants, replicating human behavior in the robot. The method deals with the problem of several interlocutors appearing and disappearing from the visual field of the robot. A robotic head has been designed and built and a virtual agent projected on the robot's face display has been integrated with the gaze control. Different experiments have been carried out with that robotic head integrated into a ROS architecture model. The work presents the analysis of the method, how the system has been integrated with the robotic head and the experiments and results obtained.

Keywords: gaze control, gaze engagement, HRI, humanoid robot, robotic head, ROS, competitive network, computer vision

1. INTRODUCTION

The gaze control of a robotic head represents an important field of research in robotics, since it promotes higher evaluations of a robot's comprehension and naturalness (Kousidis and Schlangen, 2015) in human-robot interaction. This gaze engagement represents a key factor in interaction because humans feel more comfortable if robots behave like a person. Some robots look and track people, but are not able to change between several interlocutors during a conversation. If someone disappears from the field of view, the robot listens to a sound and turns its head, looking for someone to follow. This behavior is not natural in a conversation of several people.

The method proposed in this article responds to the problem of several individuals interacting with a robotic head. It replicates human behavior using a competitive neural network which receives stimuli from each person who interacts with the robot. Different factors are taken into account: look, who speaks, pose, hoard conversation, habituation, etc. These factors produce stimuli which create a dynamic conversation, independently of whether several people appear and disappear from the visual field. Different considerations have been taken into account:

(i) There may be several people in front of the robot.

(ii) One or several people could speak at the same time.

(iii) People could enter and exit the visual field of the robot.

(iv) People could appear and disappear from the scene due to occlusions or false detections.

(v) People may be looking at the robot or elsewhere.

(vi) People must be distinguishable from each other to facilitate their monitoring.

(vii) People could request the robot's attention in several ways (entering their visual field, talking, moving in front of the robot).

(viii) The robot should give more attention to new stimuli (e.g., a person starts talking, while another one has been talking for a certain time).

(ix) People can be in different image planes (closer or farther from the robot).

(x) The transition of the change between two people should be smooth and weighted, avoiding jerky movements.

The proposed method is based on a competitive network that accurately combines a set of identification techniques, facial monitoring, and behavioral rules to achieve the most natural interaction. The system deals with the presence of different stimuli and allows a stable determination of the focus of attention that must be followed with the robot's eyes. At the same time, the system has principles of adaptation and stability. The robot must be able to respond quickly to new stimuli, but at the same time the response must be smooth and stable, avoiding erratic behaviors.

The robot's gaze control not only falls on the movement of the head. A projected virtual face, hereinafter referred to as the agent, has been created to move the eyes and show expressions based on such factors as the presence of people. As an example, if there is no one in front of the robot, the agent will show a sad expression and begin an exploratory movement. In Ishi et al. (2010), an experiment with two different robots showed a more natural behavior of one of them just because the ability to move the lips.

The present paper is structured as follows: section 2 explores the state-of-art of the technologies considered in this paper. Section 3 shows how the method works, exploring the different steps: face recognition, people pose, speaking detection, competitive network, etc. Section 4 explains how this method has been integrated with a robotic head developed for this purpose. In section 5, the different experiments and results obtained with the robotic head are reported. An overall discussion on the obtained results is stated. Finally, section 6 notes the advantages and limitations of the presented system and suggests future developments.

2. OVERVIEW OF RELATED WORK

Gaze control has been an important field of research over the last few years since it contributes to the improvement of communication with people. As stated by Kousidis and Schlangen (2015), when a robot is a listener in a multi-party conversation and tracks the conversation with its gaze, it promotes higher evaluations of that robot's comprehension and naturalness than a robot performing random gazing between speakers. Moreover, Garau et al. (2001) proposed that virtual agents which use turn taking gaze during conversations are evaluated as more natural and pleasant than agents that use random gaze or none gaze control in their communication. What is more, their conversation is rated as more engaging. Boucher et al. (2012) studied the gaze effects of Human-human interaction in a cooperation experiment and implemented a heuristic capability to generate such gaze cues by a humanoid robot. However, that work was mainly focused on the interaction with just one user. In addition, as studied by Andrist et al. (2015), the gaze behavior more effectively motivates users to repeatedly engage in therapeutic tasks.

Neurorobotics plays an important role in Human–Robot Interaction (HRI), a discipline that allows improving robots which can communicate and respond to ongoing human communications and behavior (Kiesler and Hinds, 2004). It also plays an essential role in assistive and rehabilitation robotics (Beckerle et al., 2017). Admoni and Scassellati (2017) have recently presented a survey of the state of the art in social eye gaze for HRI. The authors distinguish between three different approaches to the problem: Human-focused, centered on understanding the characteristics of human behavior during interactions with robots; Design-focused, which studies how the design of a robot impacts on interactions with humans; and Technology-focused, with the aim of researching how to build computational tools to guide the robot's gaze in human interaction. According to the authors, the main challenges of conversation are managing attention and turn-taking between partners, selecting the correct gaze for the conversational content, and adopting the right conversational roles. In addition, Thrun (2004) indicated that the shape of the robot, specifically humanoid features, influences people's behavior toward the machine and their expectations about its capabilities.

Regarding gaze control, an outstanding work was published by Zaraki et al. (2014), where the authors created a system to guide a robot's gaze at multiple humans who were interacting with the robot. The attention mechanism used features which had been proven to guide human attention. The authors relied on the use of a Kinect sensor to track people and obtain sound direction. This system considers the maximum of the sum of different elements: social features, proxemics values, orientation, and a memory component. However, the stimuli considered were limited and the maximum value could change abruptly and could also lead to erratic changes in the focus of attention. Our proposal does not need a 3D sensor and is able to work with common RGB cameras, while a competitive neural network provides soft transitions between different focuses of attention. Another remarkable work was published by Alonso-Martín et al. (2012), who used 8 different microphones in a social robot, named Maggie, to determine which direction to look. Once the orientation of the robot with respect to the user was obtained, an infrared laser returned the distance with respect to him/her to make the robot move forward/backward. That work was able to guide the robot in the direction of a speaker, but did

not consider face stimuli or a competitive behavior. A robot equipped with two laser range-finders was also used by Schulz et al. (2003) to probabilistically track the position of people with a mobile robot, although it did not have a cognitive behavior. Saldien et al. (2014) presented a robot focus on Robot Assisted Therapy (RAT) which was able to perceive different stimuli (visual, auditory, and tactile) and track a certain colored object, a face or a directional voice. However, that work did not solve the problem of who to pay attention. Vega et al. (2013) proposed a dynamic visual memory to store the 3D processed information from a moving camera on board the robot. The attention system chose where to look, according to the principles of reobserving objects in the visual memory and the need to explore new areas. The visual memory was a collection of relevant task-oriented objects and 3D segments. However, that work was mainly focused on creating a visual memory about objects and reobserving them according to the basis of keeping the memory updated. More recently, Viciana-Abad et al. (2014) have demonstrated the benefits of fusing sensory information with Bayes inference. The authors localize a person with a robotic head by simultaneously processing visual and audio data. The authors mainly focus on tracking a particular person instead of a conversation between multiple participants.

Visual information represents an important aspect of HRI, increasing the robot's awareness. The robots must trade off features which affect the utility of the visual information (Gergle et al., 2013), such as the robot's field of view (FOV), alignment of perspective, degree of spatial resolution, frame rate or synchronization with a voice stream. The Visual Focus of Attention (VFOA) represents who or what people are looking at. Massé (2018) presents a VFOA model based on a Bayesian network to infer the relation between head poses and object locations. That work exploits the concept of correlation between eye gaze and head movements instead of using face landmarks. The author uses a convolutional neural network to predict object locations and a reinforcement learning method for robotic gaze control. The robot autonomously learns a strategy for moving its head using audio and visual observations. The author mainly focuses on the relation between head poses and objects and not the conversations between different participants. In Ghiţă et al. (2018), the authors track people in a robot assistive care scenario by using an *Oriented FAST and Rotated BRIEF* detector (ORB) and comparing characteristics between frames. According to their study, NAOqi functions, which are SO/API for Pepper, NAO and Romeo robots, are improved, covering more range, situations of occlusions and more orientations of a person/face. Shiomi et al. (2004) develop a face to face tracking of people, generating hypotheses about people position by using peripheral vision. Even though a face may not be present in the foveal vision of the robot when it is gazing at another object, the robot keeps plausible hypotheses about the location of the human faces. However, in Csapo et al. (2012), the authors point out some problems with Nao platform about non-verbal human-robot interaction, where some capabilities of detecting faces or tracking people interfere with other modules that send commands to the same motor, producing senseless movements due to conflicting signals.

Regarding the identification of people, the feature-based systems play an important role in both human and robotics perception (Potapova et al., 2017). Different techniques have been developed for face detection and recognition during the last years. As stated by Zafeiriou et al. (2015), robust feature extraction methodologies have been used for face detection, such as *Scale Invariant Feature Transform* (SIFT) features (Lowe et al., 1999; Geng and Jiang, 2009; Lenc and Král, 2015), *Histograms of oriented Gradients* (HoGs) (Dalal and Triggs, 2005), *Local Binary Patterns* (LBPs) (Ahonen et al., 2006), or *Haar cascade classifiers* (Viola and Jones, 2004). Among the most advanced techniques, Haar classifiers, HoG detector or Deep Learning based solutions are widely used, as they are implemented in OpenCV or DLIB libraries. Haar classifiers detect faces at different scales but do not work with non-frontal faces and occlusions and return a large number of false predictions. The HoG feature descriptor is fast but does not detect small faces (less than 80×80 pixels). It can work with some minor occlusions or non-frontal faces, but returns a bounding box that often excludes part of the forehead/chin. The DLIB library (King, 2009) implements a CNN face detector using a Maximum-Margin Object Detector (King, 2015), which works for different face orientations and occlusions. However, it does not return a precise bounding box real-time. OpenCV offers a DNN Face Detector, based on a Single-Shot-Multibox detector (Liu et al., 2016), which is very accurate, works with different face orientations, scales and occlusions, and runs in real-time on CPU. Another important aspect of face recognition is the extraction of the face's features. They are mainly obtained using an *Active Appearance Model* (Cootes et al., 2001; Milborrow and Nicolls, 2008) or DLIB-68 model (Kazemi and Sullivan, 2014), which makes a face alignment with an ensemble of regression trees before obtaining the corresponding landmarks. Finally, the face recognition is mainly based on Deep Residual Learning algorithms, which are very accurate. DLIB implements a ResNet network with 29 convolution layers and uses a pre-trained model which takes the 68 face landmarks obtained from an image (Kazemi and Sullivan, 2014).

As in the case of facial recognition, human body recognition relies on the use of *Haar filters* (Viola et al., 2001), HoG (Dalal and Triggs, 2005), or *Deep Convolution Neural Networks* (DCNN), such as the *Faster RCNN Inception V2 COCO Model* (Ren et al., 2015). All of them are available in OpenCV, working simultaneously with TensorFlow. Systems based on DCNN offer better results, detecting more people with a lower number of false positives.

Lip activity detection has also been studied in different works. Bendris et al. (2010) proposed a method to detect lip motion by measuring the degree of disorder of pixel directions around the lip using the optical flow technique (Saenko et al., 2005). A rectangle around the lips was enlarged to be aligned with a previous one by taking the region that minimized the mean squared difference (MSD). Siatras et al. (2008) considered the increased average value and standard deviation of the number of pixels with the lowest intensities of the mouth region to detect visual speech. They created a statistical algorithm that used two detectors based on noise to characterize visual speech and silence in video sequences.

FIGURE 1 | Steps in the gaze control process.
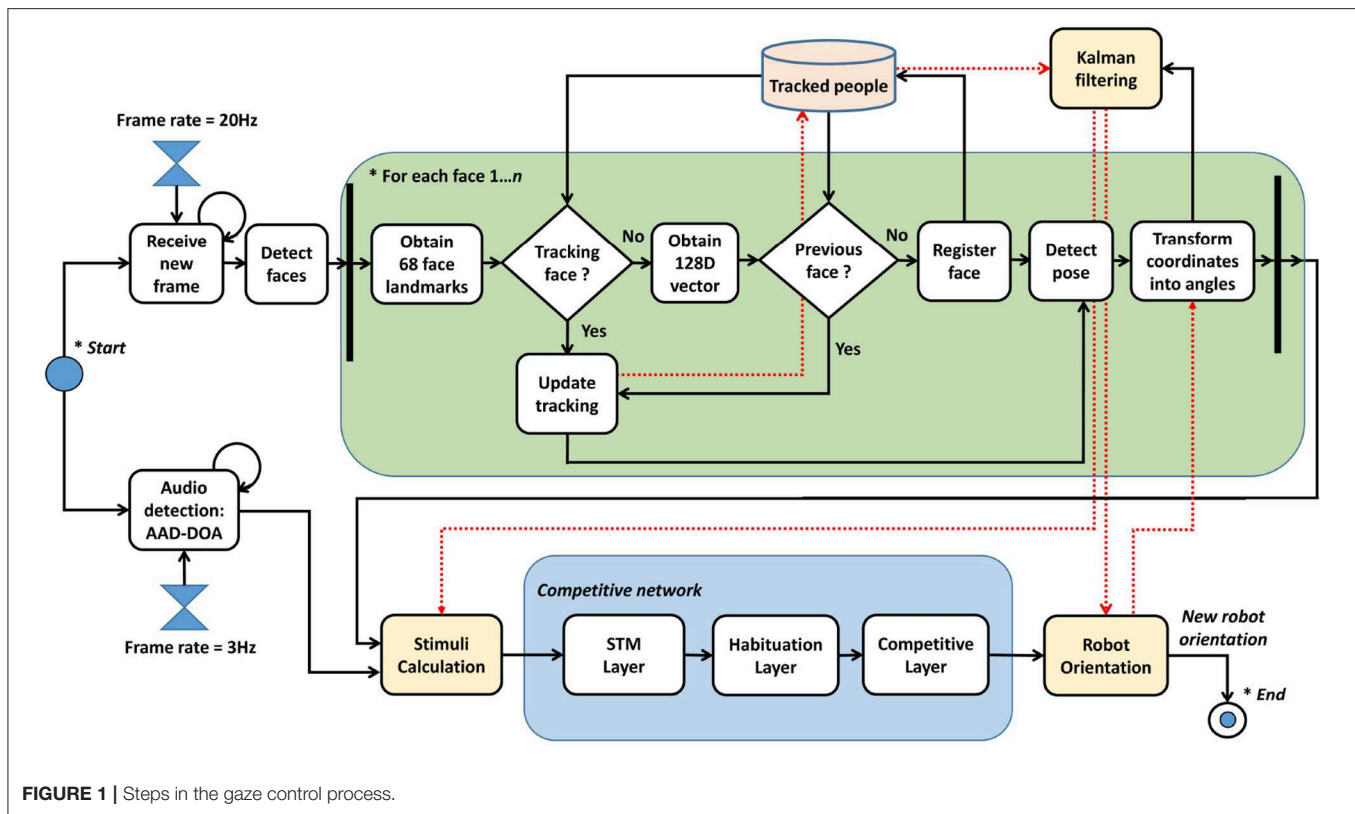
Neural networks have played a major role in the interaction of robots with people. An outstanding work was presented by Bicho et al. (2010), who presented a control architecture for human-robot collaboration which was formalized by a coupled system of dynamic neural fields representing a distributed network of neural populations that encode in their activation patterns goals, actions, and shared task knowledge. This approach was valid for inferring the response to user stimuli but was only valid for the collaboration with one user. In our paper, a competitive network is used to create a dynamic behavior in the interaction between several participants. A competitive network consists of a layer that is able to react to different stimuli and decide a winner. This network has a progressive behavior which does not switch sharply between consecutive winners. A habituation layer avoids a participant from being the winner for a long period of time when a new, different stimulus arrives. For example, imagine that two people are talking and one of them is monopolizing the conversation. The robot would gaze at the person who monopolizes the conversation but, if the other person says something, a more natural behavior is to gaze at the new interlocutor.

3. ANALYSIS OF THE SYSTEM

The method proposed in this section analyzes how the gaze control of a robotic head works. **Figure 1** shows the different steps needed to obtain the orientation angles of the robotic head.

There is a specific frame rate that determines how many images are received per second. Whenever a frame is received, there is a process of people detection and face landmarks are obtained using the DLIB 68 model (Kazemi and Sullivan, 2014). There is a temporal table which memorizes the people who have interacted with the robot. If a person is new, a 128D vector is obtained by techniques related to facial recognition, as explained later. This vector allows a concrete person to be identified. Once a person has been identified, the method uses correlation tracking (Danelljan et al., 2014) to follow the person during consecutive frames. If a person disappears from the scene for a while, the method is able to keep their 128D vector in its memory to make a subsequent re-identification. Different stimuli, such as visual speaking detection and pose estimation, are evaluated for each person, and they represent the entry of a competitive network composed of three different layers:

- *Short Time Memory layer (STM)*, which extends the duration of the concrete stimuli. As an example, if a person says a short sentence, the system might not properly consider the stimulus. This layer extends the duration of the stimuli to have more value in the entry of the habituation layer.
- *Habituation layer*, which penalizes persistent stimuli against new stimuli, e.g., it prevents someone from hoarding the conversation. When a person is speaking continuously, if someone says something, it makes it easier to pay attention to the new interlocutor.

- *Competitive layer*, which is the final step of the network. The input to the competitive network includes stimuli for each person in the memory, and decides who is the winner. This layer is responsible for following a dynamic behavior, without abrupt jumps.

Kalman filters are used to process the position of people over time. These filters have been previously used on robotic heads, such as Milighetti et al. (2011), who predicts the next state of the moving target. In our work, 2D face positions obtained during tracking are transformed into concrete angles of the robotic head (ψ_c and θ_c). These angles are integrated in different Kalman filters, one for each person in memory, giving the position of the person independently of being in the robot's FOV. After a winner is provided by the competitive network, the new orientation of the robotic head is that returned by its Kalman filter. Then, the Kalman filters are updated with the last positions obtained.

The rest of this section is composed of different subsections: section 3.1 explains the different stimuli considered and how they are calculated. Section 3.2 presents how the competitive network works. Section 3.3 shows how the face coordinates of a person obtained from an image are transformed into the angles of the robot. Finally, section 3.4 presents how the Kalman filter is used.

3.1. Entry Stimuli of Competitive Network

Each person, k, produces a set of stimuli, x. This set of stimuli are introduced into the competitive network and defined I_{kx}. Several stimuli that consider how people react in a conversation have been used. The stimuli that can be present or absent are coded as binary values and are balanced in importance by a weight, w_{kx}:

- I_{k1} is the stimulus associated to a person k who is situated in the robot's field of view (FOV). People situated in front of the robot are candidates to be interacting with the robot. w_{k1} is the corresponding weight associated to that stimulus.
- I_{k2} is the stimulus associated to a person k who is considered to be speaking. Lip movement detection is performed, based on mouth landmarks. Moreover, in order for a person to be considered as a speaker, incoming audio has to be detected in its direction.
- I_{k3} is the stimulus associated to a person k who is gazing directly at the robot. The pose is an important stimulus which indicates that a person is visually interacting with the robot. This stimulus represents the *mutual gaze*, a kind of *shared looking* which is related to the increase of the engagement in the interaction (Sidner et al., 2004).
- I_{k4} is the stimulus associated to a person k who is continuously moving. In a conversation with several people, an individual tends to look at another restless person. This stimulus requires the individual to be situated in the FOV of the robot. If the sum of differences of a person's position between several frames is over a concrete threshold, the person is considered to be restless.
- I_{k5} is the stimulus associated to a person k who is not situated in the robot's FOV, but for whom audio has been detected. When a person is interacting with a group of people and someone is speaking at their left/right side, this individual tends to turn the head in that direction looking for the person.

- I_{k6} is the stimulus associated to a person k who is not situated in the robot's FOV, but who is the VFOA of another group of people. When two or more people in the FOV are gazing in the same direction, a stimulus is given to people in that direction. In this situation, this stimulus considers the Visual Focus of Attention (VFOA).
- I_{k7} is the stimulus associated to a person k who is situated at a certain distance, following a proxemic approach as in other works (Alonso-Martín et al., 2012; Zaraki et al., 2014). This stimulus is multiplied by an adjustment factor which depends on the distance between the person and the robot.

Next, the calculus of each stimulus is explained.

3.1.1. Stimulus I_{k1}: Person in the FOV of the Robot

The robot detects faces in frames and persons are, first of all, recognized and labeled. When the robot turns its head, if a new face appears on the FOV, the robot can discern whether this person was previously recognized. When a face is in the FOV during a period of time, re-identification is not required since the face is followed by tracking.

There are different techniques for face recognition, but those based on Deep Residual Learning are very accurate. The library DLIB implements a ResNet network with 29 convolution layers. This model is similar to the ResNet-34 network (He et al., 2016) with a few layers removed and the number of filters per layer reduced by half. This library uses a pre-trained model and takes the 68 face landmarks obtained from an image (Kazemi and Sullivan, 2014), aligns the face and maps it to a 128 dimensional vector space where images of the same person are close in terms of distance. Once the 128D vector has been obtained, the similarity of two faces is calculated, checking if their Euclidean distance is small enough. Using a threshold of 0.6, the DLIB model obtains an accuracy of 99.38% on the standard LFW face recognition benchmark (Huang et al., 2008). This procedure requires the use of GPU, due to the fact that obtaining the 128D vector from a face takes more than 0.4 s with CPU. With an intel i9-9900K and a GPU GeForce RTX 2080 Ti, the 128D vector is obtained in less than 0.05 s.

When a person is identified, a tracker is used to follow the person in the FOV. It increases the speed of the system, as tracking is faster than face recognition. The DLIB library allows a correlation tracker based on Danelljan et al. (2014) to be used. This method considers the approach of Bolme et al. (2010) and makes use of learning discriminative correlation filters based on a scale pyramid representation. The authors use separate filters for tracking, in real-time, objects that change in both translation and scaling.

3.1.2. Stimulus I_{k2}: Person Speaking in the FOV of the Robot

As stated in the overview of related work, there are different works which have studied lip activity to discern whether a person is speaking or not. Speaking detection is important in a conversation because, when a person is looking at the robot but is in silence, it is likely that another person who is speaking receives the robot's attention.

FIGURE 2 | Visual speaking detection. **(A)** Points used for lip activity detection. **(B)** Points to normalize distance.
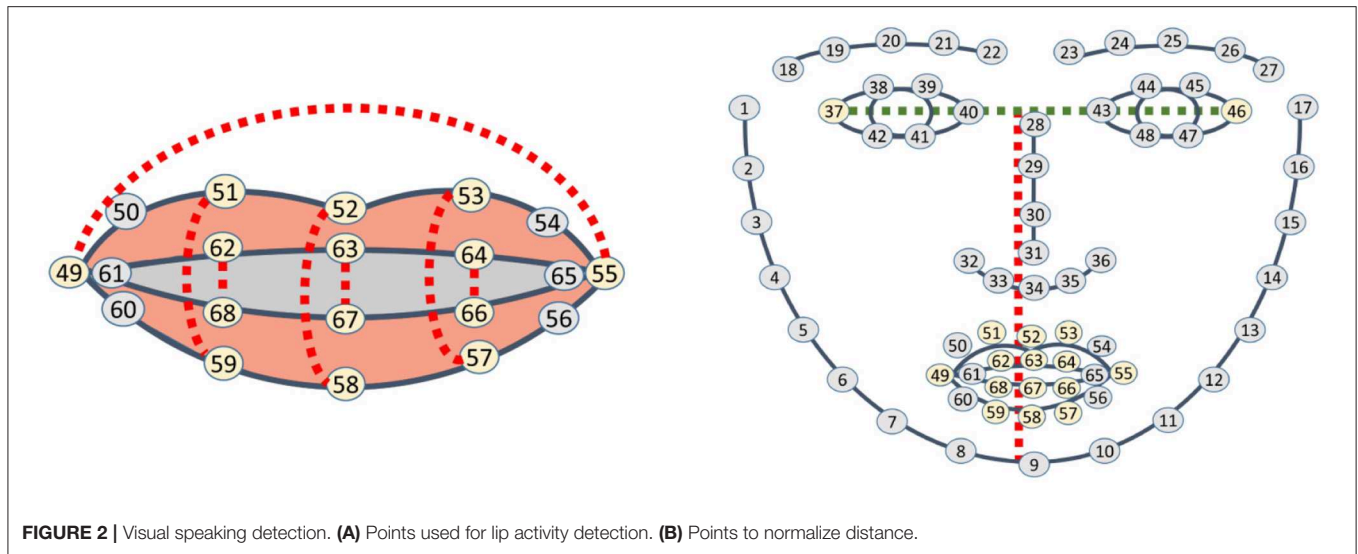
Instead of implementing a complete analysis of images as in Bendris et al. (2010) or Siatras et al. (2008), and due to the fact that DLIB 68 returns characteristic points of the mouth (points 49–68), the movement between lips is calculated in consecutive frames using these points. The sum of distances between a group of points (62 and 68, 63 and 67, 64 and 66, 49 and 55, 51 and 59, 52 and 58, 53 and 57) (see **Figure 2A**) is divided by the distance between the middle point of the eyes, points 37 and 46, and the end of the chin, point 9 (see **Figure 2B**). This division is done to normalize the distance, regardless of whether the person is close or far away.

Let d_k be the result of the calculus of the separation of lips in frame k, as shown in Equation (1).

$$
d_k = \frac{\left|\overrightarrow{P_{62}, P_{68}}\right| + \left|\overrightarrow{P_{63}, P_{67}}\right| + \left|\overrightarrow{P_{64}, P_{66}}\right|}{\left|\overrightarrow{P_m(P_{37}, P_{46}), P_9}\right|}
$$
$$
+ \frac{\left|\overrightarrow{P_{49}, P_{55}}\right| + \left|\overrightarrow{P_{51}, P_{59}}\right|}{\left|\overrightarrow{P_m(P_{37}, P_{46}), P_9}\right|}
$$
$$
+ \frac{\left|\overrightarrow{P_{52}, P_{58}}\right| + \left|\overrightarrow{P_{53}, P_{57}}\right|}{\left|\overrightarrow{P_m(P_{37}, P_{46}), P_9}\right|} \tag{1}
$$

d_k is added during a few consecutive frames, say 5 frames. If the result is over a threshold τ_S, that is $\sum_{k=1}^{5} d_k \geq \tau_S$, the person is visually considered to be speaking.

After the visual speaking confirmation, it is necessary to evaluate if there is audio in the direction of that person. A person can move his/her lips, i.e., breathing, but in silence. An Audio Activity Detection (AAD), combined with a Direction of Arrival system (DOA) (Griffin et al., 2012), is used to detect the zone where audio is originated (left, central, right). When the person is visually speaking in the robot's FOV and audio is detected in

the direction of the robot's gaze (80°), that person is considered to be speaking.

3.1.3. Stimulus I_{k3}: Person Gazing Directly at the Robot

The determination of people pose is an important stimulus because, when a person is looking at the robot, there is a greater interaction between both participants: person and robot.

Instead of developing a complete analysis of image, such as Ba and Odobez (2008), where the authors study how to link head position with the visual focus of attention, modeling the pose observations with a Gaussian Mixture Model (GMM) or a Hidden Markov Model (HMM), the use of known face landmarks is exploited. A Perspective-n-Points algorithm (PnP) associates 2D points of the DLIB 68 model (Kazemi and Sullivan, 2014) with 3D points in a respective model. Using a standard 3D model of a head, with some characteristic points such as nose tip, chin, left eye left corner, right eye right corner, mouth left corner or mouth right corner, it is possible to calculate the respective pose between the DLIB points and the 3D model. PnP is implemented in different ways, but the solution DLT + Levenberg-Marquardt Optimization has been chosen. A *Direct Linear Transformation* (DLT) algorithm allows L, the projection matrix of the camera, to be calculated. In the formula $LX_i = u_i$, 2D coordinates u_i are related to a 3D point X_i. To obtain L, the six different points selected (nose tip, chin, etc.) and the 3D position of these points in the model are introduced in the algorithm. The Levenberg-Marquardt Optimization finds a pose that minimizes the re-projection error, which is the sum of the squared distances between the observed image points and the projection.

When the pose has been determined, and considering the nose tip as the origin of coordinates, a 3D vector from the origin is projected onto the 2D image: say $x_1(0, 0, 0) - x_2(0, 0, 500)$ projected as $u_1(x_1, y_1) - u_2(x_2, y_2)$. The module of this vector, $\left|\overrightarrow{u_1, u_2}\right|$, which is represented in blue in **Figure 3**, is divided by the distance between the middle point of the eyes, points 37 and 46, and the end of the chin, point 9 (red vector). This division
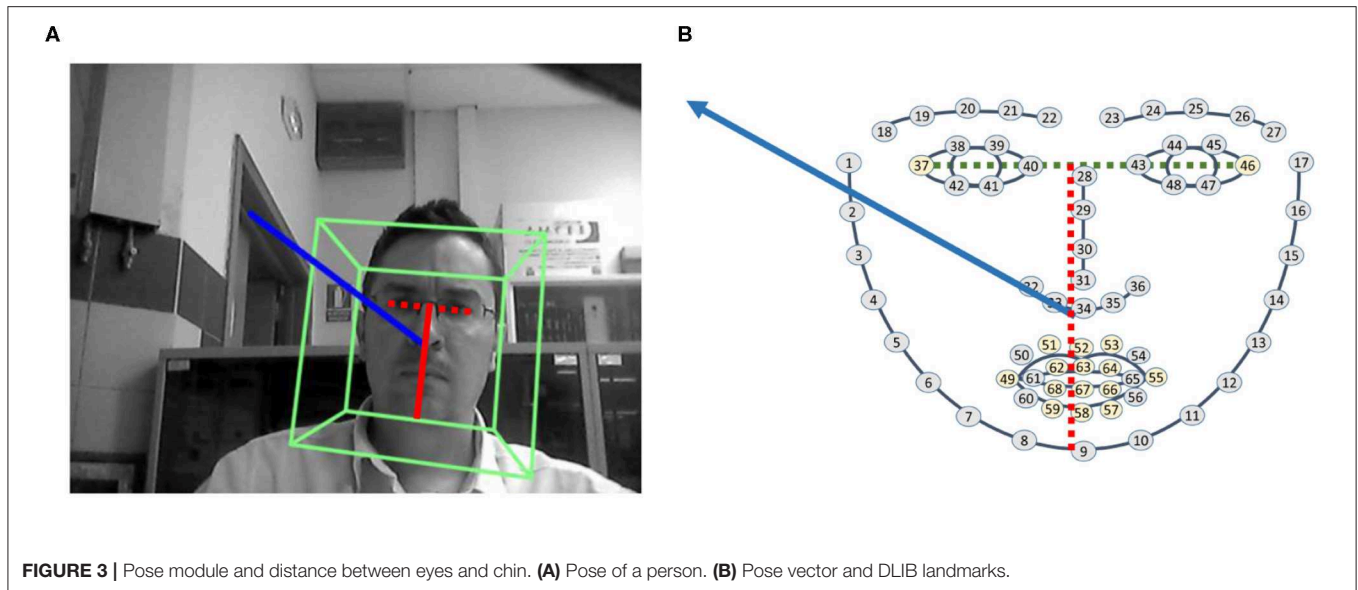
FIGURE 3 | Pose module and distance between eyes and chin. **(A)** Pose of a person. **(B)** Pose vector and DLIB landmarks.

normalizes the module, making it independent of the distance to the face. If the result is below a given threshold, $\tau_M < 5$, the person is considered to be looking at the robot and their stimulus is increased in the entrance of the competitive network.

3.1.4. Stimulus I_{k4}: Person Continuously Moving

In a conversation with several people, a person tends to look at another restless one. The angles associated to a person, ψ_c and θ_c, are stored during 25 consecutive frames, as this number of frames has produced the most accurate results. The difference between each pair of consecutives angles is calculated and accumulated. These differences are computed using the Euclidean distance. If the accumulated distance is over a concrete threshold, τ_M, the person is considered to be a restless person.

3.1.5. Stimulus I_{k5}: Person Not in the Robot's FOV but With Audio

In a conversation with several people, a person has to turn the head left/right when other persons are speaking in that direction. As explained before, the proposed method uses Kalman filters to keep the last estimated position of each person in memory. The DOA system (Griffin et al., 2012) indicates the direction of audio: left, central, right. When nobody is speaking in the robot's FOV, all people situated to the left/right side of the robot, according to the person's angles and robot pose, receive stimulus if audio has been detected in their zone.

In addition, there are two fictional persons who are situated in the left/right zone of the robot, respectively. When a new participant begins speaking, who has not been previously detected by the robot, the corresponding fictional person receives stimulus whenever audio is detected in their zone and nobody is speaking in the robot's FOV. This is used to integrate that person in the competition and be able to get the robot's attention.

The DOA system is also used when the robot has not previously detected any person. When a sound arrives from a

concrete direction, the robot will begin an exploratory movement in that zone to search for people.

3.1.6. Stimulus I_{k6}: Person in the VFOA of Other People

As assumed by other authors (Massé, 2018), it is important to consider the *Visual Focus of Attention* (VFOA). It represents who or what people are looking at. If two persons situated in the robot's FOV are looking at someone who is situated to the left/right of the robot (see **Figure 4A**), there is probably a reason and the robot should consider looking in that direction.

To calculate this stimulus, if two or more people in the robot's FOV are not looking at the robot and there is a difference between their pose vector below a given threshold, say $30°$, the stimulus is increased for the people who are situated in the direction of the gaze of these persons.

3.1.7. Stimulus I_{k7}: Proxemics of a Person

Proxemics is the study of the human use of space. As stated by Hall et al. (1968), there are four distinct zones in the interpersonal relations: (1) intimate space, (2) personal space, (3) social space, and (4) public space. Following this proxemic approach, as in other works (Alonso-Martín et al., 2012; Zaraki et al., 2014), the distance between the robot and a person is estimated. In each frame, this distance is proportional to the distance between the middle point of the eyes, points 37 and 46, and the end of the chin, point 9. A function that approximates this distance has been obtained, as follows:

$$dp_k = -0.02 \cdot \left| \overrightarrow{P_m(P_{37}, P_{46}), P_9} \right| + 2.04 \tag{2}$$

where dp_k represents the distance between the robot and a person k in meters. Stimuli increased are those of people situated at the
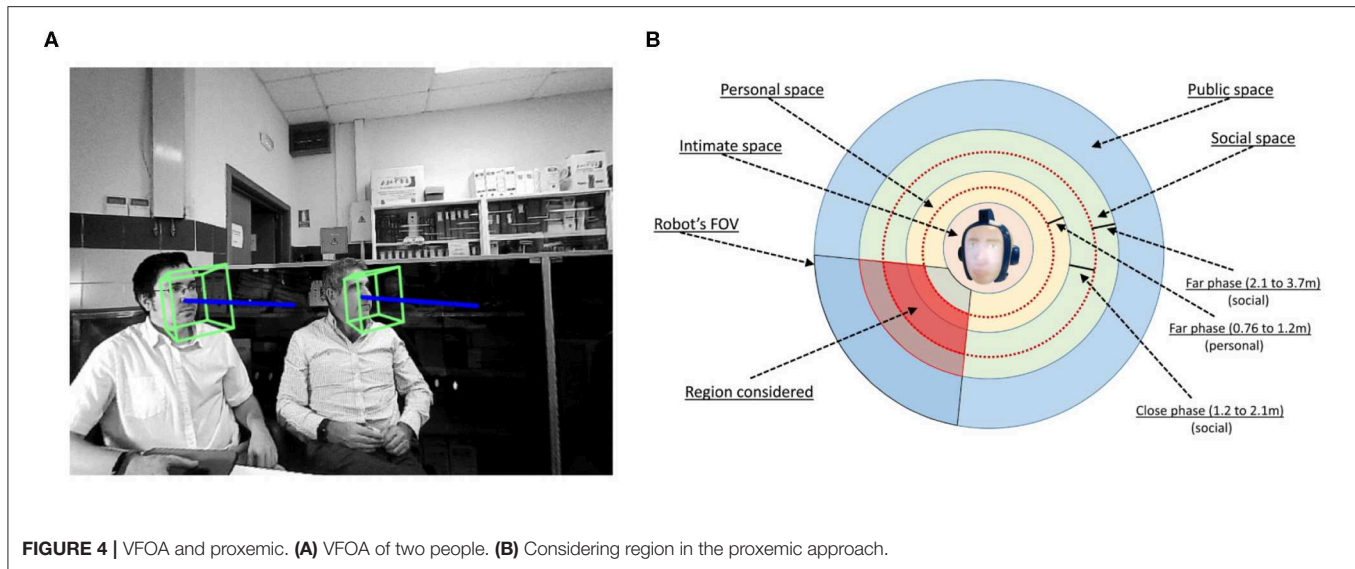
FIGURE 4 | VFOA and proxemic. **(A)** VFOA of two people. **(B)** Considering region in the proxemic approach.

far phase of the personal space (0.76–1.22 m) or at the social space (1.22–3.70 m), as shown in **Figure 4B**.

This stimulus, I_{k7}, is multiplied by an adjustment factor depending on the distance:

$$f_k = \begin{cases} 1 & if \quad dp_k >= 0.76 \ \& \ dp_k <= 2.10 \\ \frac{3.70-dp_k}{1.60} & if \quad dp_k > 2.10 \ \& \ dp_k <= 3.70 \end{cases} \quad (3)$$

People situated at the *far phase* of the personal space (0.76–1.22 m) or at the *close phase* of the social space (1.22–2.10 m), multiply their stimulus by $f_k = 1$. People situated at the *far phase* of the social space (1.22– 3.70 m) multiply their stimulus by a factor, f_k, which depends on the distance and is reduced until $dp_k = 3.70m$. These equations have been adjusted during experiments.

3.2. Competitive Network

When all stimuli have been obtained, a competitive network decides which person to gaze at. The competitive network creates a dynamic behavior between interlocutors. Instead of changing the gaze between participants abruptly, the network softens the change and avoids situations such as monopolizing attention. It has three components: an STM layer, which increases the duration of short stimuli; a Habituation layer, which penalizes persistent stimuli against the novel ones through a dynamic gain; and a Competitive layer, which creates a competition between participants and decides a winner.

This network has one input and output for every possible person, as seen in **Figure 5**. Thus, I_{ij} corresponds to the stimulus j for the person i and w_{ij} to its associated weight. The weights take values between 0 and 1 and are chosen experimentally according to the relevance of the stimuli. The selected person is the one with the highest value in the output, O_i. The network is composed of three interconnected layers.

One important aspect in this process is the configuration of the different parameters of such a neural network. The weights of the different stimuli have to be computed in order to achieve

behavior similar to humans. During an initial training, three people follow a list of steps previously recorded. At the same time, another person observes the interaction and annotates the time instants when a person should be the focus of attention. When all data have been obtained, stimuli from expected winners and losers are separated based on the said manual annotation. The process is modeled as an optimization problem, maximizing the sum of the distances between the winners and the losers at each time instant t, as shown in Equation (4). This procedure ensures that the weights are optimal to make the selected persons winners and separate them from the losers.

$$\max \sum_{t=1}^{m} \left(\sum_{k \in losers} I_{t,winner} - I_{tk} \right)$$

$$= \max \sum_{t=1}^{m} \left[\sum_{k \in losers} \left(\sum_{x=1}^{7} w_x \cdot I_{t,winner,x} - \sum_{x=1}^{7} w_x \cdot I_{t,k,x} \right) \right] \quad (4)$$

3.2.1. Short Time Memory Layer

An STM increases the duration of the stimulus to have more value in the entry of the habituation layer. It is based on the model proposed by Grossberg (1982). The neuron activity is computed using Equation (5), where x_i is the activity of the neuron i and A_1 is the decay rate. The next term is the auto-reinforcement, which makes neuron activity tend to its saturation value B_1. C_1 marks the growth rate. S_i is the filtered stimulus and w_i is the STM weight for that stimulus.

$$\frac{dx_i}{dt} = -A_1 x_i + C_1(B_1 - x_i)[S_i w_i] \quad (5)$$
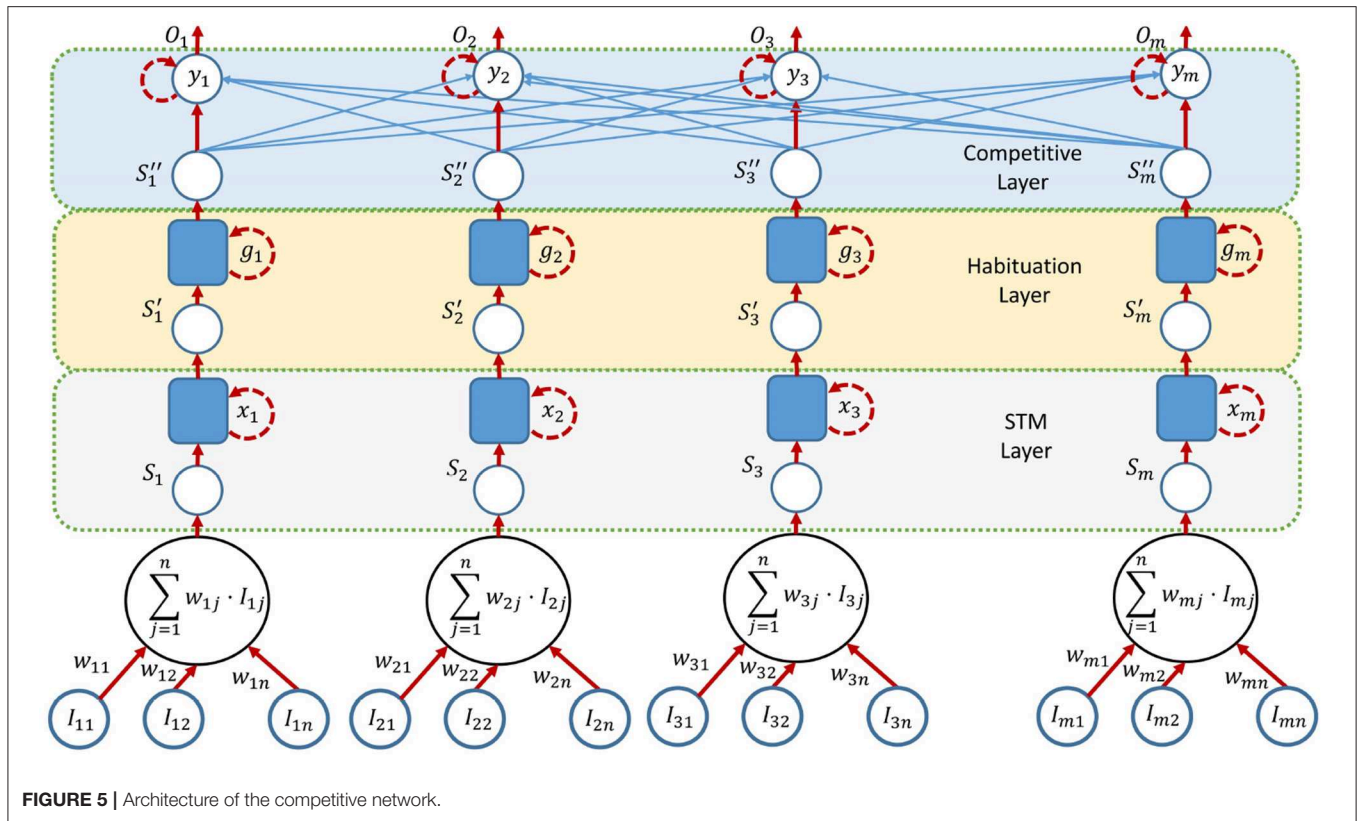
FIGURE 5 | Architecture of the competitive network.

Equation (5) is solved in real time using a trapezoidal integration defined by the following equations:

$$x_i(kh) = x_i((k-1)h) + \frac{g_i(kh) + g_i(k-1)h}{2} \tag{6}$$

$$g_i(kh) = -A_1 x_i(kh) + C_1 x_i(kh) + C_1(B_1 - x_i(kh))[S_i w_i] \tag{7}$$

where k represents an increasing value $0..n$ and h is the period of time (0.1 in experiments). **Figure 6A** reflects the behavior of the STM layer with parameters $A_1 = 0.2$, $B_1 = 1$, and $C_1 = 0.5$. When a short stimulus arrives, the layer maintains its value during some time until it completely disappears.

3.2.2. Habituation Layer

The proposed method has habituation capabilities, that is, it loses interest in permanent stimuli over time. Habituation networks were proposed by Grossberg (1968). In this layer, stimuli decay due to habituation allows the network to dynamically adapt against permanent inputs, such as a person who is hoarding the conversation. By using a habituation layer, those continuous stimuli lose preponderance over time, allowing novel stimuli to acquire more importance. Habituation is carried out by multiplying the input stimuli with a dynamic gain that is updated over time. The gain g_i computation is calculated based on Equation (8).

$$\frac{dg_i}{dt} = E(1 - g_i) - FS_i' g_i \tag{8}$$

In the same way as with the STM layer, the differential equation is integrated trough the following numerical discrete equations:

$$g_i(kh) = g_i((k-1)h) + \frac{p(kh) + p((k-1)h))h}{2} \tag{9}$$

$$p(kh) = E[1 - g_i(kh)] - FS_i'(kh)g_i(kh) \tag{10}$$

where S_i' is the filtered stimulus and g_i is the habituation gain for that stimulus. When a stimulus is active, the habituation gain decreases from the maximum value of 1 to a minimum value given by $E/(E + FS_i')$, proportional to the stimulus value S_i'. This gain is recharged to its initial unity value when the stimulus ends. Charge and discharge rates are determined by the parameters E and F. **Figure 6B** shows the behavior of the Habituation layer with parameters $E = 0.5$ and $F = 0.02$. When there is a long stimulus in duration, the layer decreases its value to give more possibilities to other new stimuli from other people.

3.2.3. Competitive Layer

The outputs of the habituation layer are the inputs of the competitive layer. The competitive model used is on-center off-surround (Grossberg, 1982) and is based on the model of Hodgkin (1952), where each neuron is reinforced with its own activity, but attenuated by the activity of the neurons it is connected to. This attenuation is known as lateral inhibition and makes people compete by means of the weight of their corresponding stimulus. Neuron activity is computed using Equations (11) and (12), where y_i is the activity of neuron i and A_2
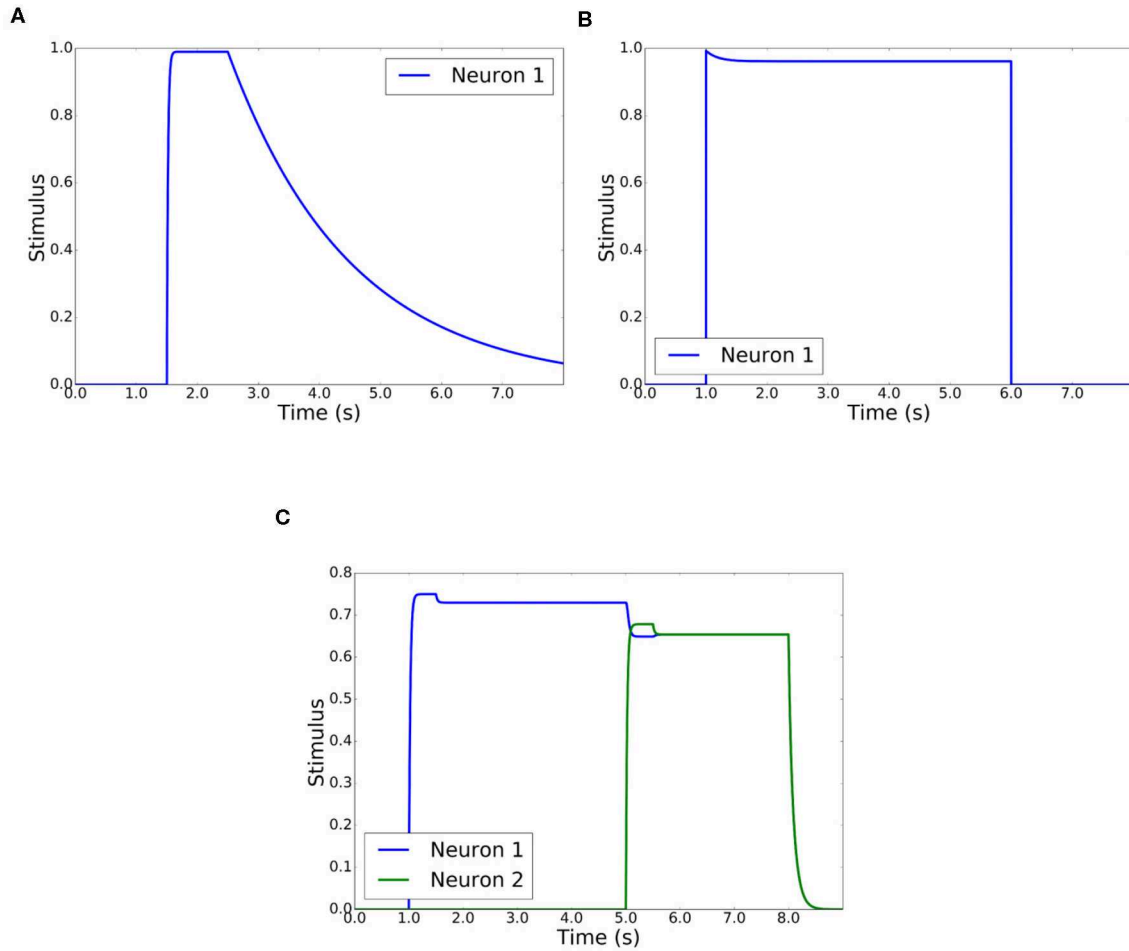
FIGURE 6 | STM, habituation, and competitive layer behavior. **(A)** STM layer behavior when there is a short duration stimulus. $A1 = 0.2$, $B1 = 1$, and $C1 = 0.5$. **(B)** Habituation layer behavior when there is a long duration stimulus. $E = 0.5$, $F = 0.02$. **(C)** Behavior of competitive layer with 2 neurons. $A2 = 1$, $B2 = 1$, and $C2 = 3$, $D = 1$, $E = 0.5$, and $F = 0.02$.

is the decay rate. The next term is the auto-reinforcement, which makes neuron activity tend to its saturation value B_2. C_2 marks the growth rate and $S_i^{''}$ is the filtered stimulus. Finally, the last term represents lateral inhibition (off-surround).

$$\frac{dy_i}{dt} = -A_2 y_i + C_2(B_2 - y_i)[S_i^{''} + f(y_i)]$$
$$-y_i \sum_{i \neq j} f(y_j) \tag{11}$$
$$f(y_i) = Dy_i^2 \tag{12}$$

As before, the differential equation is integrated trough the following numerical discrete equations:

$$y_i(kh) = y_i((k-1)h) + \frac{q_i(kh) + q_i(k-1)h}{2} \tag{13}$$
$$q_i(kh) = = -A_2 y_i(kh) + C_2 y_i(kh)$$
$$+ C_2(B_2 - y_i(kh))[S_i^{''} w_i]$$

$$-\sum_{i \neq j} Dy_j((k-1)h)^2 \tag{14}$$

A parabolic function has been selected for $f(y_i)$, so the winner neuron is reinforced against the rest. The competition schema is a winner-take-all, as it is desirable that only one person prevails over the complementary ones. **Figure 6C** shows the behavior of the competitive and habituation layers. The competitive layer has parameters $A_2 = 1$, $B_2 = 1$, $C_2 = 3$, and $D = 1$, while the habituation layer has $E = 0.5$ and $F = 0.02$.

There are two inputs of stimuli. One input receives a stimulus of 0.7 that remains in time from $t = 1$ until $t = 8$. Another input receives the same stimulus, 0.7, from $t = 5$ until $t = 8$, where both stimuli disappear. The sequence of winners is: neuron 1 and neuron 2. Neuron 1 is the first winner, as it is the only stimulus. Because of the habituation layer, neuron 1 reduces its value and neuron 2 becomes winner, although the input stimuli are similar.
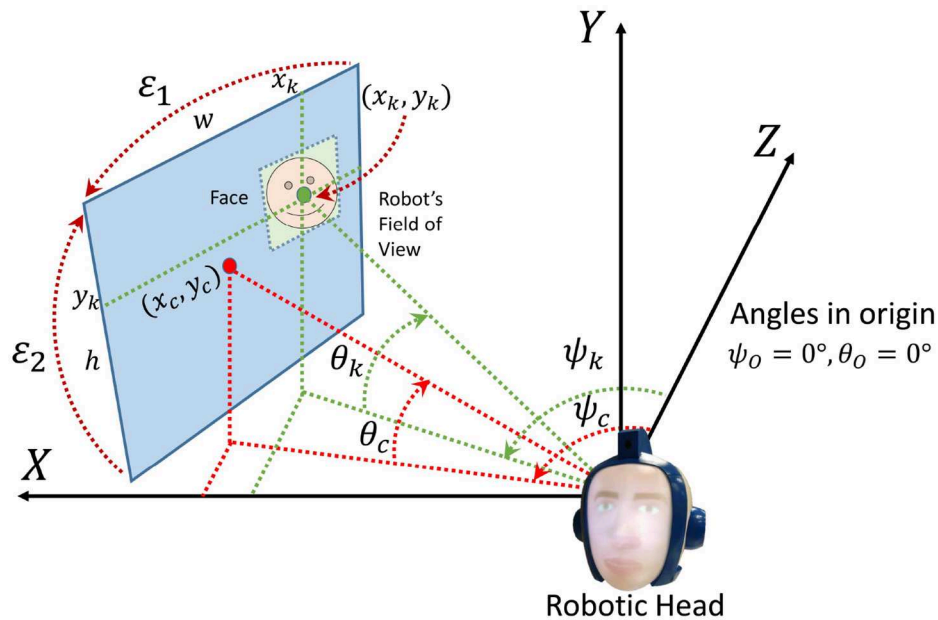
FIGURE 7 | Transformation of the coordinates of a face into angles of the robot.

3.3. Angles of the Robot

An important step of the method is the transformation of the coordinates of a face into angles of the robot. These angles are situated in a universal coordinate system (UCS) and they remain in the robot's memory regardless of whether it turns its head. Kalman filters also use these angles to estimate the position and the robot accepts angles to move its head up/down and left/right.

Initially, the robotic head is situated in angles $\psi_O = 0°$ and $\theta_O = 0°$. When it turns its head, ψ_c and θ_c represent the center point of the image. This image is the robot's FoV. As seen before, DLIB returns a rectangle for each face detected in an image. The midpoint of that rectangle is (x_k, y_k) and corresponds to the person k.

From (x_k, y_k), it is necessary to calculate the angles (ψ_k, θ_k). For that purpose, some fixed parameters are necessary. As seen in **Figure 7**, ε_1 represents the vertical angle of the FoV, while ε_2 corresponds to the horizontal one. w and h are the respective width and height of the image.

Equation (15) show the relation between (x_k, y_k) and (ψ_k, θ_k).

$$\begin{cases} \psi_k = \left[\psi_c - \frac{\varepsilon_1}{2} \right] - \left[x_k \cdot \frac{\varepsilon_1}{w} \right] \\ \theta_k = \left[\theta_c - \frac{\varepsilon_2}{2} \right] - \left[y_k \cdot \frac{\varepsilon_2}{h} \right] \end{cases} \quad (15)$$

Initially $\psi_c = \psi_O = 0°$ and $\theta_c = \theta_O = 0°$. When the robot turns its head, the target angles are aligned with the center of the image (x_c, y_c). Therefore, the robotic head moves to (ψ_k, θ_k) and, when it finishes, ψ_c takes the value ψ_k and θ_c takes the respective θ_k. At that moment, (ψ_c, θ_c) are the current situation of the robot until the next movement.

3.4. Kalman Filter

The Kalman filter and the Extended Kalman filter (Rosales and Sclaroff, 1998) are well-known techniques that allow the position

of a person to be estimated by means of positions known and previously updated. The Kalman filter considers noise and other inaccuracies and helps estimate the location. It uses Bayesian inference and estimates a joint probability distribution over the variables for each instant of time.

Several Kalman filters run simultaneously, one for each person previously tracked by the robot. When a new person appears, the proposed method considers an initial state vector with the first known positions of the person. This vector is composed of the corresponding angles of the robot calculated from the center point of the person's face, $x_k = (\psi_k, \theta_k)$. The Kalman filters are created with a posteriori error covariance matrix $P_k = \begin{vmatrix} 1 & 0 \\ 0 & 1 \end{vmatrix}$ (a measure of the estimated accuracy of the state estimate).

An iteration is produced when the person has been detected in a new frame and new positions/angles arrive. At that moment, the Kalman filter is updated with the new angles. H_k is the observation model which maps the true state space into the observed space. Specifically, the observation model is $H_k = \begin{vmatrix} 0.1 & 0 \\ 0 & 0.1 \end{vmatrix}$, and the covariance of the observation noise is $R_k = \begin{vmatrix} 0.25 & 0 \\ 0 & 0.25 \end{vmatrix}$

4. ROBOT CONSTRUCTION AND METHOD IMPLEMENTATION

A complete robotic head has been designed and built, as shown in **Figure 8**. The design considers the principle by which people will collaborate more naturally and easily with humanoid robots as compared with machine-like robots (Hinds et al., 2004), and the idea of the anticipated acceptance of a social robot when

FIGURE 8 | Robotic head developed.

it provides more enjoyable interactions (de Graaf et al., 2019). This robotic head includes two servomotors for the orientation (ψ and θ angles) and a wide angle camera located at the top of the head. An ESP32 module has been programmed to move the servomotors, which are also connected to a step-down voltage regulator. The ESP32 module is connected to a computer, which is responsible for the robot's gaze control. The function developed in the ESP receives the target where the head has to be moved as ψ and θ angles. It moves the servos to that concrete position step by step and, each step, publishes the current position of the head.

A projector situated at the back projects an agent on a 3D printed display representing the robot's face. The agent follows an approach using the *Facial Action Coding System* (FACS) (Ekman, 1997), which is a well-known method for measuring and describing facial behavior. In FACS, several *Action Units* (AUs) are responsible for contracting groups of muscles in face changes.

The computer in charge of processing the gaze control is an intel i9-9900K, with 32Gb of RAM and a GPU GeForce RTX 2080 Ti. It is connected to different elements: the ESP32 module, a mini projector with HDMI, and the circular microphone array, concretely a ReSpeaker Mic Array v2.0. This computer runs the *Robot Operating System* (ROS) (Quigley et al., 2009) over Linux, where the proposed method has been deployed. Several independent nodes run simultaneously and different messages are published and subscribed by these nodes. The architecture of the developed system is shown in **Figure 9**.

There are several nodes:

- *framePublisher*, which directly receives frames using OpenCV and publishes into a compressed image topic. It has a rate of 20 Hz. The image is transformed into a grayscale image and the dimensions are adjusted to an optimal value (500 pixels wide).
- *audioDetection*, which detects audio activity (AAD) and the direction (DOA). This node publishes a message,

audioDetection, with a rate of 3 Hz, indicating both values. Other nodes with a different rate, such as *stimuliCalculation*, use the last received values, remaining in time until other, new ones are published. Our experimentation showed that this technique produced a better behavior in the recognition of a person speaking.

- *peopleTracking*, which is responsible for different activities:

  (i) Extracting people/faces/face landmarks from a frame using DLIB libraries.
  (ii) Tracking people using face identification and a correlation tracker.
  (iii) Obtaining people pose.
  (iv) Keeping in memory a list of existing people, as well as their position, depending on the angles of the robotic head. The competitive network has 20 entries corresponding to possible interlocutors during the last activity of the robot. If the list is complete and a new person appears, the person not detected for the longest time is replaced by the new one.
  (v) Based on the list of existing people, publishing a message, *faceLandmarks* topic, with face characteristics (68 DLIB points, coordinates, pose) for the stimuli calculation and a message, *peoplePosition* topic, with new known positions for the Kalman filters.

- *stimuliCalculation*, which calculates the value of the entries of the competitive network using the information provided by the *faceLandmarks* topic. The stimulus i can take the values 0 or 1 for a person k and is multiplied by a weight w_i with the values represented in **Table 1**.

  All stimuli multiplied by their corresponding weights are accumulated for each person k, being normalized between 0 and 1, and they are published as a *stimuliVector* topic, which is the input of the competitive network.

  This node performs another important task: when nobody is speaking in the robot's FOV, whenever there is audio in the left or right zone, the stimulus I_5 is increased for a fictional person. There are two fictional persons who, respectively, have Kalman filters with angles in the left/right zone of the robotic head. This task is valid for situations where a new person appears to the left/right of the robot. As they have not been previously recognized by the robot, it does not take them into account. The use of the competition network includes them in the dynamic behavior process.

- *competitiveNetwork*, which implements the three previously explained layers:

  (i) *Short Time Memory layer (STM)*, which extends the duration of concrete stimuli.
  (ii) *Habituation layer*, which penalizes persistent stimuli against novel ones.
  (iii) *Competitive layer*, which decides the winner.

  This node publishes a message with the winner, *winner* topic, which is in fact the stimuli vector processed by the network. The output with the highest value represents the winner.

- *Kalmanfilter*, which implements a Kalman filter for each different person. It receives the angles of the robot head for
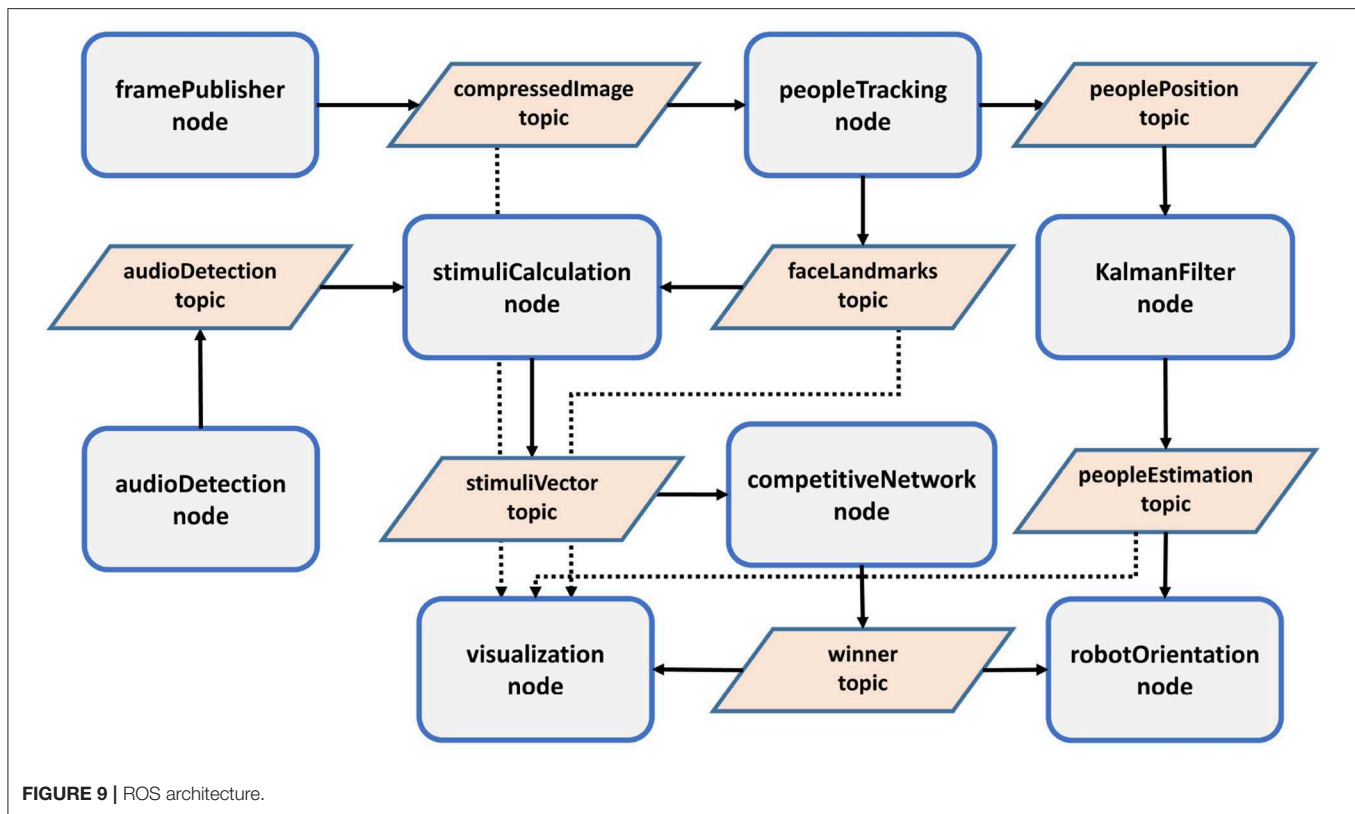
FIGURE 9 | ROS architecture.

TABLE 1 | Weights in stimuli calculations (obtained with the optimization problem).

Stimulus i	l_{ki}	w_i
1. Person k detected in the robot's FOV	1/0	0.06
2. Person k speaking in the robot's FOV	1/0	0.25
3. Person k gazing directly at the robot	1/0	0.06
4. Person k moving sharply between frames	1/0	0.16
5. Person k not in the robot's FOV but with audio	1/0	0.25
6. Person k in the VFOA of other people	1/0	0.16
7. Proxemics of person k	1/0	0.06

tracking a person, who is in the robot's FOV, and updates their corresponding values.

- *robotOrientation*, which is in charge of combining information received from the competitive network and the estimation of the Kalman filter, *peopleEstimation* topic, and sends the new angles to the robotic head. Another task is carried out by this node: when nobody has been detected during the last 20 s, the robot begins an exploratory movement whenever new audio is detected. The exploration initially moves the attention to the zone where the audio has been detected (left, central, right). During this exploration, the agent modifies the robot's expression from neutral to sad.

    Based on the winner of the competition layer, the *robotOrientation* node obtains the target angles (ψ_k and θ_k) of the winner k, from the Kalman filter. Before sending them

to the ESP32 module, it makes some controls related to the agent movements:

(i) If the distance between (ψ_k, θ_k) and (ψ_{robot}, θ_{robot}), the current position of the robotic head, is lower than 10°, the node does not send the movement to the servos. Instead, a movement of the agent's eyes is produced in the direction of the target. This step avoids servo gittering.

(ii) If that distance is over 10°, the node sends the movement to the servos and, at the same time, makes the agent smile and move the eyes in the direction of the target.

It is also interesting to mention that a blinking eyes movement has been given to the robot to show realism.

- *visualization*, which visually displays the results of the process on the computer screen.

5. EXPERIMENTS AND RESULTS DISCUSSION

An initial step consisted in determining the optimal parameters of the system. This step was carried out by means of the optimization problem previously described, where three people followed a list of established actions. An external observer followed the interaction and annotated the time instants when a person should be the focus of attention. The complete list of actions had 726 steps and the optimization problem was solved in 18 iterations and 0.23 s (in an intel i9-9900K) using the SLSQP algorithm (Kraft and Schnepper, 1989). The obtained results

FIGURE 10 | Robot interaction with two people. **(A)** Input stimuli in a conversation. **(B)** Output stimuli in a conversation. **(C)** Input stimuli with different actions. **(D)** Output stimuli with different actions.
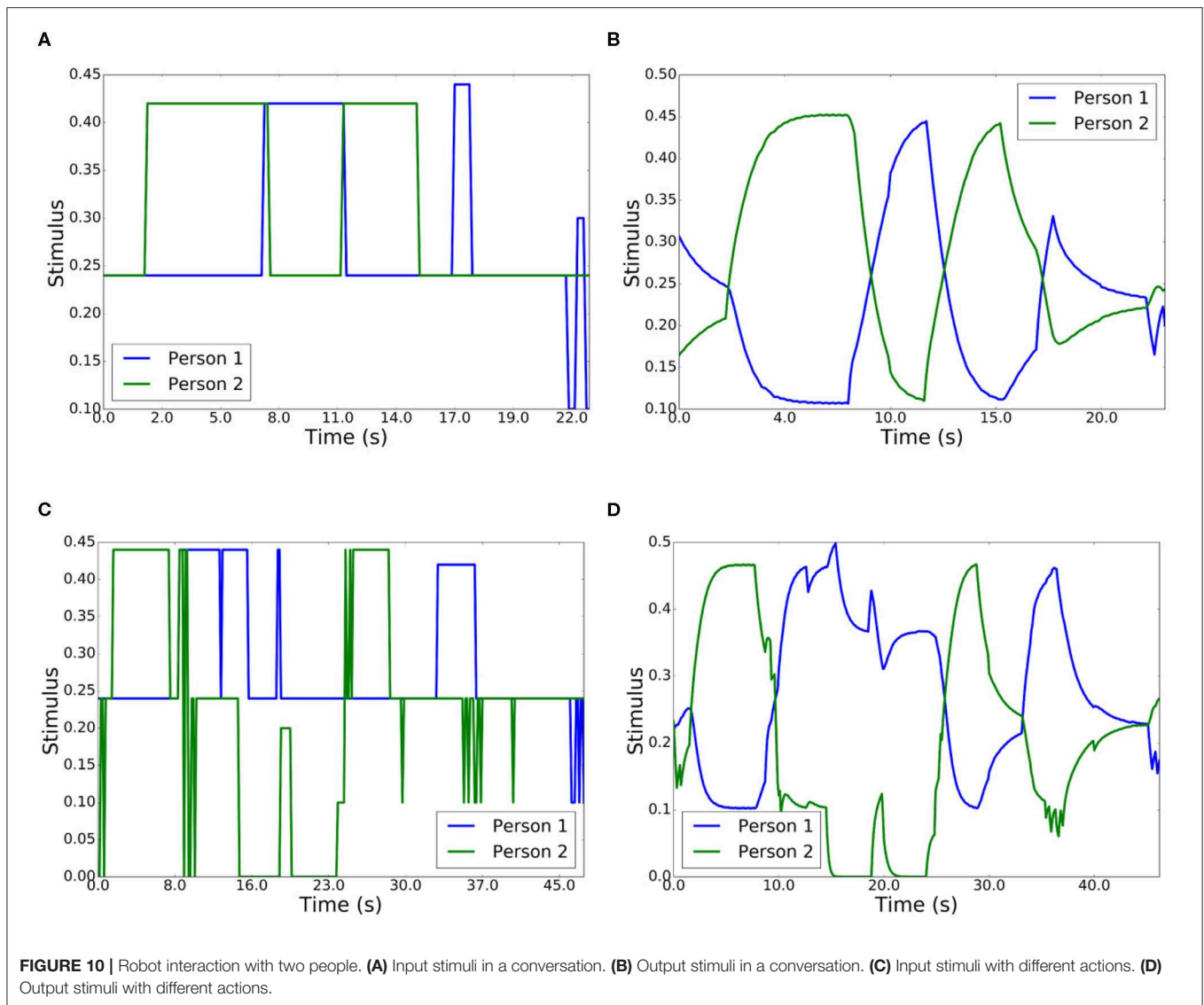
were $w_1 = 0.06$, $w_2 = 0.25$, $w_3 = 0.06$, $w_4 = 0.16$, $w_5 = 0.25$, $w_6 = 0.16$, and $w_7 = 0.06$. It is important to mention that, depending on the behavior expected by the observer, some parameter may take slightly different values. After this step, several situations involving two/three interlocutors and the robotic head were considered. 13 interactions (with 31.200 frames) with two or three people were recorded, considering different stimuli. 8 people participated in the experiments repeating some of them with similar results. The results of 4 out of them are shown next.

During the first experiment, two people situated in the social space interacted with the robot. **Figure 10A** represents the summation of input stimuli in the competitive network, while **Figure 10B** shows the output. In this experiment, the two persons looked at the robot and had a conversation. Person 2 began the conversation (2–7 s), followed by 1 (6–12 s), 2 (12–15 s), and finally 1 (17–18 s). At the end of the conversation, the habituation

capabilities appeared and person 2 became the winner (22–24 s). The output of the competition redirected the gaze of the robot dynamically.

During the second experiment with two people, a preset sequence of people's behavior was evaluated, as shown in **Table 2**. Depending on the stimuli obtained from both participants, the result of the competition was dynamically modified. This table shows the interval of time over which the stimulus takes place. The results of the winning person are shown in **Figures 10C,D**.

The following experiments were carried out with three people, as seen in **Figure 11**. In the case **Figure 11A**, there are two individuals in the robot's FOV. There is a conversation between three people on the scene and the robot moves its head depending on the winner, but, at a specific time, the stimuli produced by person C (speaking detection, gazing, proxemics, movement, etc.) and the result of the competitive network makes this person the winner and the robot turns its head to look at him/her.

The second case, **Figure 11B**, is similar to the first situation but the three individuals are in the robot's FOV. The robot moves its head or eyes as soon as a new winner is elicited from the competitive network. As before, when person C becomes the winner, the robot turns its head right to center the person. The third case, **Figure 11C**, corresponds to a situation where the robot had previously detected three people. At a certain time, nobody is in the FOV, but the robot is detecting audio. The

competitive network returns winner A and the robot turns its head left, according to the values returned by the Kalman filter associated to that person.

The fourth case, **Figure 11D**, represents two persons who are quiet in the robot's FOV. A new person arrives and begins to speak to the left. As soon as the robot detects sound in that zone, the fictional input of the competitive network corresponding to the left zone receives new stimulus. The competition evolves until that fictional person becomes the winner and, finally, the robot turns its head in that direction.

Figures 12A,B show the input and output values of the competition for the speaking interaction between three people. These three people were gazing at the robot and situated at the *close phase* of the social space (1.22–2.10 m). Person 3 began the conversation (10–17 s), followed by 1 (17–23 s), 2 (23–32 s), 3 (32–38s), 1 (38–42 s), 2 (42–48 s), 3 (48–53 s), 1 (53–60 s), and finally, 2 (60–63 s). During the seconds 35-38, person 3 was hoarding the conversation. However, person 1 began to speak and became the winner due to the habituation capabilities.

Table 3 shows, as in the case of two people, a preset sequence of people's behavior. **Figures 12C,D** show the corresponding input and output values for the competition, where several

TABLE 2 | Sequence of behavior of two people.

State	Person speaking		Person gazing robot		Person moving		Winner	Time
	1	2	1	2	1	2		
1		●	●	●			2	2–10 s
2	●		●	●			1	10–16 s
3			●				1	16–23 s
4		●	●	●			2	23–33 s
5			●	●	●		1	33–37 s
6				●			2	45–48 s

FIGURE 11 | Different situations interacting with the robotic head. **(A)** Three people in the FOV. **(B)** Two people in the FOV. **(C)** Nobody in the FOV. **(D)** A new person appear on scene.
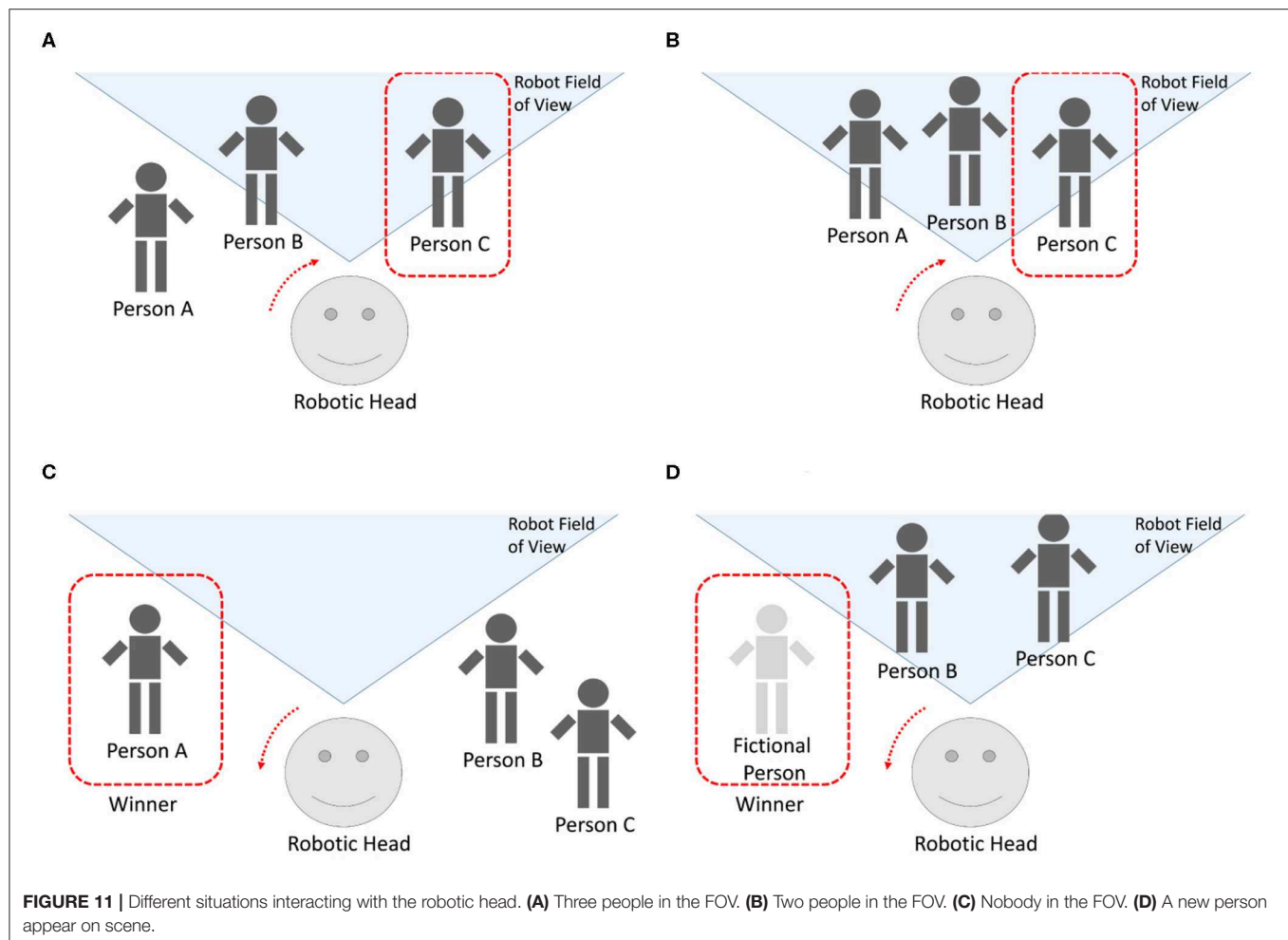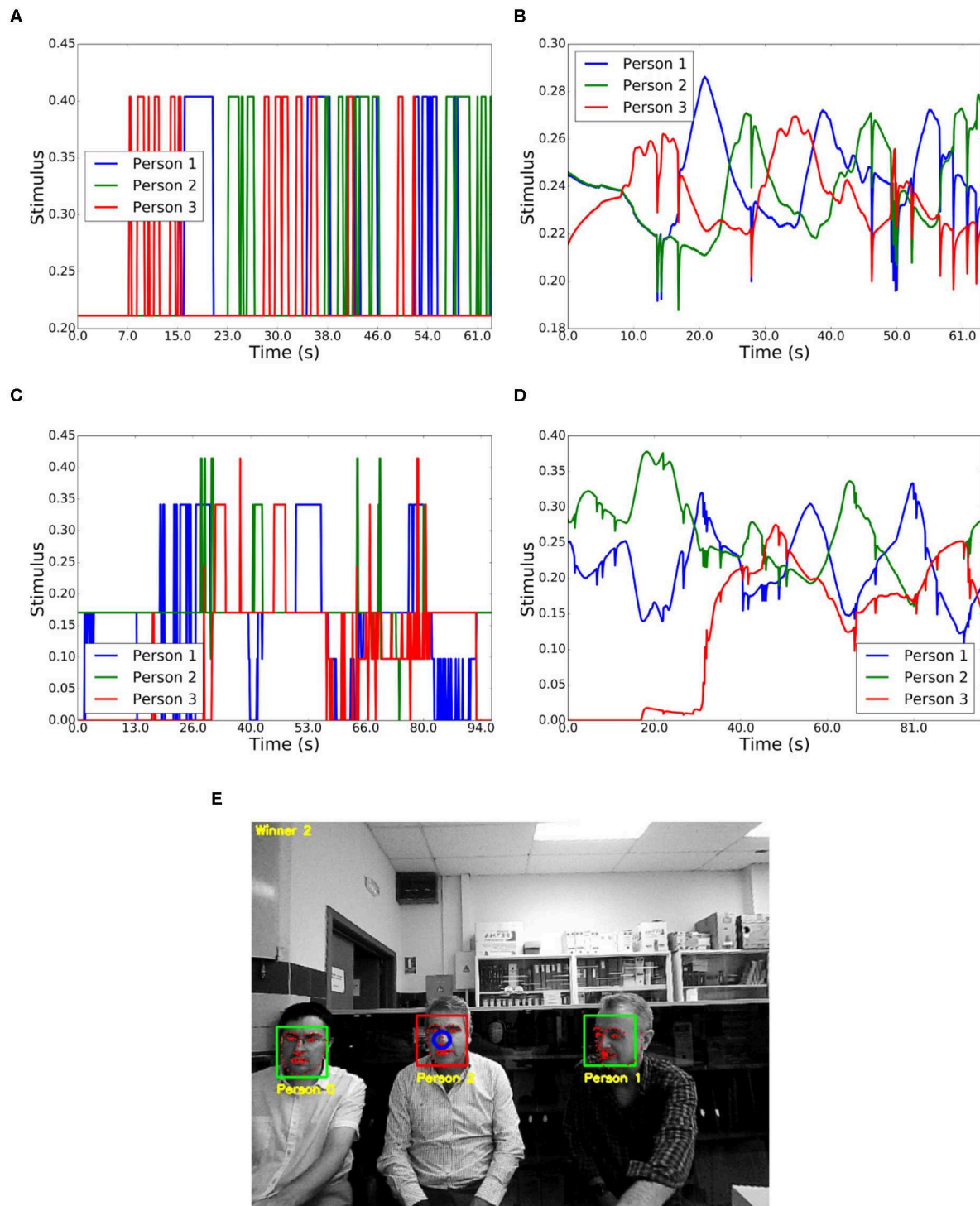
FIGURE 12 | Interaction between three people. **(A)** Input stimuli in a conversation. **(B)** Output stimuli in a conversation. **(C)** Input stimuli with different actions. **(D)** Output stimuli with different actions. **(E)** Result of the *visualization* node showing the winner.

TABLE 3 | Sequence of behavior of three people.

State	Detected/Proxemics			Gazing at robot			Moving			Speaking			Winner	Time
	1	2	3	1	2	3	1	2	3	1	2	3		
1	●	●			●								2	0–18 s
2	●	●	●	●	●								2	18–22 s
3	●	●	●	●	●		●						1	22–28 s
4	●	●	●	●	●			●		●			2	28–42 s
5	●	●	●		●				●				3	42–52 s
6	●	●	●		●	●	●						1	52–62 s
7	●	●	●		●						●		2	62–75 s
8	●	●	●	●								●	1	75–83 s
9	●	●	●		●	●							3	83–94 s
10	●	●		●	●								2	94–100 s

stimuli were considered, such as gazing at the robot, proxemics, speaking or moving. **Figure 12E** shows a moment of the experiment which is displayed by the *visualization* node.

The proposed system reflects a successful result of 85.0% in relation to the behavior of a person in the same situation. To make this estimation, the result of the interaction has been supervised by comparing the selection of the focus of attention proposed by the system (winner nodes) and the one that would have been determined by a person observing the scene. The failures found were mainly due to errors in the perception of stimuli such as external audio noises unrelated to the interaction, and blurred images due to the effect of the movement of people and the robot itself. It is difficult to make a comparison of the results with other authors since there are no common datasets and each author uses different sensors and stimuli. A research work close to ours is that of Zaraki et al. (2014), where a precision between 75.2 and 89.4% is obtained depending on saccadic and non-saccadic movements. However, the conditions of the experiments are very different. In that work, a Kinect RGB-D and a DIK-ABLIS eye tracking system are used. In our experiments, a common webcam and an inexpensive ReSpeaker Mic array V2.0 are used. Moreover, in Zaraki et al. (2014) the interaction between 2 people at a certain distance is analyzed prioritizing gestural and postural acts, while our experimentation has been carried out at a short distance with the participation of three interlocutors (in fact, the number of interlocutors is not limited in the proposed method) and prioritizing facial and audio stimuli (where the people are looking, who is speaking, who is moving, their distance to the robot and even if someone gets out of the FOV).

Finally, mention that the program is available on the Internet at the URL:
https://github.com/jaiduqdom/robotGazeControl.git.

6. CONCLUSIONS

This work presents a system to control the gaze of a robot interacting with multiple people in conversations. Several computer vision techniques have been used to obtain a set of stimuli, which are received by a competitive network that decides a winner and indicates where to look.

Different types of weighted stimuli have been considered, allowing the robot to focus its attention on one interlocutor. Thus, the proposed system identifies the interlocutors and re-identifies them when they leave the robot's FOV, tracks them using correlation tracking and Kalman filters, determines whether the person is speaking by identifying lip movement in concurrence with a sound source, studies where people are looking, monitors the person's movements to see if they wish to attract attention, and prioritizes close interlocutors over those further away. Kalman filters keep the angular positions of each particular person who has previously interacted with in the robot's memory, regardless of whether they are in the FOV or not.

The competitive network, by means of an adequate weighting of weights, allows who the robot should look at in order to create an interaction to be determined. The characteristics of the network allow a smooth transition between the focus of attention through the competition of stimuli. An important characteristic is the habituation mechanism that tends to prioritize new stimuli over existing ones and prevents a certain user from monopolizing the focus of attention.

Different experiments have been carried out in which two and three people have interacted with each other and with the robot. The results show how the competition of the network executed in real time allows a behavior of the robot similar to a person when choosing the focus of attention to be obtained.

A robotic head has been designed and built to evaluate the system, where a virtual agent projected on a 3D printed display has been used to represent the robot's face. Depending on the response of the gaze control system, the agent has shown different expressions and movement of the eyes. A ROS-based architecture has been presented and the different experiments carried out have been detailed.

The robot's behavior looks natural and is perceived similar to that of humans. The method has shown itself to be an important improvement in robot gaze control, creating a more realistic HRI system which is more acceptable to interlocutors than other

robots that turn their heads without a dynamic and human-oriented method. The response of the competitive network has succeeded in producing soft transitions between different focuses of attention.

Finally, it can be noted that: people need to be properly detected by the face recognizer, which requires they are situated within a given distance range from the robot (2.2 m in our experiments); a GPUs is required for real-time processing (i.e., face recognition takes 0.4 s with CPU and 0.05 s with GPU in our experiments); and complexity of the statistical data analysis to perform inferential tests may differ from one person to another, due to inherent criteria differences.

The future objectives of the project will be the development of a conversational system, providing speech and voice recognition capabilities, and the perception and generation of emotions. These components, together with the developed gaze control system, will offer a low-cost, intelligent robot with human-like behavior. Simultaneously, the method will be available for other existing robots, since the ROS architecture means it can be integrated with other types of robots.

DATA AVAILABILITY STATEMENT

The program, a **Video S1** and the experimental datasets (ROS badges) are available on the Internet at the URL: https://github.com/jaiduqdom/robotGazeControl.git.

ETHICS STATEMENT

Written informed consent was obtained from the individual(s) for the publication of any potentially identifiable images or data included in this article.

AUTHOR CONTRIBUTIONS

JD-D, JG-G-B, and EZ conceived, designed and performed the experiments, analyzed the data, and wrote the paper.

FUNDING

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fnbot.2020.00034/full#supplementary-material

Video S1 | Interaction with the robotic head.

REFERENCES

Admoni, H., and Scassellati, B. (2017). Social eye gaze in human-robot interaction: a review. *J. Hum. Robot Interact.* 6, 25–63. doi: 10.5898/JHRI.6.1.Admoni

Ahonen, T., Hadid, A., and Pietikainen, M. (2006). Face description with local binary patterns: application to face recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* 12, 2037–2041. doi: 10.1109/TPAMI.2006.244

Alonso-Martín, F., Gorostiza, J. F., Malfaz, M., and Salichs, M. A. (2012). User localization during human-robot interaction. *Sensors* 12, 9913–9935. doi: 10.3390/s120709913

Andrist, S., Mutlu, B., and Tapus, A. (2015). "Look like me: matching robot personality via gaze to increase motivation," in *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (Seoul: ACM), 3603–3612. doi: 10.1145/2702123.2702592

Ba, S. O., and Odobez, J.-M. (2008). Recognizing visual focus of attention from head pose in natural meetings. *IEEE Trans. Syst. Man Cybernet. B* 39, 16–33. doi: 10.1109/TSMCB.2008.927274

Beckerle, P., Salvietti, G., Unal, R., Prattichizzo, D., Rossi, S., Castellini, C., et al. (2017). A human-robot interaction perspective on assistive and rehabilitation robotics. *Front. Neurorobot.* 11:24. doi: 10.3389/fnbot.2017.00024

Bendris, M., Charlet, D., and Chollet, G. (2010). "Lip activity detection for talking faces classification in TV-content," in *International Conference on Machine Vision* (Hong Kong), 187–190.

Bicho, E., Louro, L., and Erlhagen, W. (2010). Integrating verbal and nonverbal communication in a dynamic neural field architecture for human-robot interaction. *Front. Neurorobot.* 4:5. doi: 10.3389/fnbot.2010.00005

Bolme, D. S., Beveridge, J. R., Draper, B. A., and Lui, Y. M. (2010). "Visual object tracking using adaptive correlation filters," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (San Francisco, CA: IEEE), 2544–2550. doi: 10.1109/CVPR.2010.5539960

Boucher, J.-D., Pattacini, U., Lelong, A., Bailly, G., Elisei, F., Fagel, S., et al. (2012). I reach faster when I see you look: gaze effects in human-human and human-robot face-to-face cooperation. *Front. Neurorobot.* 6:3. doi: 10.3389/fnbot.2012.00003

Cootes, T. F., Edwards, G. J., and Taylor, C. J. (2001). Active appearance models. *IEEE Trans. Pattern Anal. Mach. Intell.* 6, 681–685. doi: 10.1109/34.927467

Csapo, A., Gilmartin, E., Grizou, J., Han, J., Meena, R., Anastasiou, D., et al. (2012). "Multimodal conversational interaction with a humanoid robot," in *2012 IEEE 3rd International Conference on Cognitive Infocommunications (CogInfoCom)* (Kosice: IEEE), 667–672. doi: 10.1109/CogInfoCom.2012.6421935

Dalal, N., and Triggs, B. (2005). "Histograms of oriented gradients for human detection," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)* (San Diego, CA: IEEE). doi: 10.1109/CVPR.2005.177

Danelljan, M., Häger, G., Khan, F., and Felsberg, M. (2014). "Accurate scale estimation for robust visual tracking," in *British Machine Vision Conference* (Nottingham: BMVA Press). doi: 10.5244/C.28.65

de Graaf, M. M., Ben Allouch, S., and van Dijk, J. A. (2019). Why would i use this in my home? A model of domestic social robot acceptance. *Hum. Comput. Interact.* 34, 115–173. doi: 10.1080/07370024.2017.1312406

Ekman, R. (1997). *What the Face Reveals: Basic and Applied Studies of Spontaneous Expression Using the Facial Action Coding System (FACS).* Oxford University Press.

Garau, M., Slater, M., Bee, S., and Sasse, M. A. (2001). "The impact of eye gaze on communication using humanoid avatars," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Seattle, WA: ACM), 309–316. doi: 10.1145/365024.365121

Geng, C., and Jiang, X. (2009). "Face recognition using sift features," in *2009 16th IEEE International Conference on Image Processing (ICIP)* (Cairo: IEEE), 3313–3316. doi: 10.1109/ICIP.2009.5413956

Gergle, D., Kraut, R. E., and Fussell, S. R. (2013). Using visual information for grounding and awareness in collaborative tasks. *Hum. Comput. Interact.* 28, 1–39.

Ghiță, Ş. A., Barbu, M.-Ş., Gavril, A., Trăscău, M., Sorici, A., and Florea, A. M. (2018). "User detection, tracking and recognition in robot assistive care scenarios," in *Annual Conference Towards Autonomous Robotic Systems* (Bristol, UK: Springer), 271–283. doi: 10.1007/978-3-319-96728-8_23

Griffin, A., Pavlidi, D., Puigt, M., and Mouchtaris, A. (2012). "Real-time multiple speaker DOA estimation in a circular microphone array based on matching pursuit," in *2012 Proceedings of the 20th European Signal Processing Conference (EUSIPCO)* (Bucharest: IEEE), 2303–2307.

Grossberg, S. (1968). Some nonlinear networks capable of learning a spatial pattern of arbitrary complexity. *Proc. Natl. Acad. Sci. U.S.A.* 59, 368–372. doi: 10.1073/pnas.59.2.368

Grossberg, S. (1982). "Contour enhancement, short term memory, and constancies in reverberating neural networks," in *Studies of Mind and Brain* (Springer), 332–378. doi: 10.1007/978-94-009-7758-7_8

Hall, E. T., Birdwhistell, R. L., Bock, B., Bohannan, P., Diebold, A. R. Jr., Durbin, M., et al. (1968). Proxemics [and comments and replies]. *Curr. Anthropol.* 9, 83–108. doi: 10.1086/200975

He, K., Zhang, X., Ren, S., and Sun, J. (2016). "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (Las Vegas, NV), 770–778. doi: 10.1109/CVPR.2016.90

Hinds, P. J., Roberts, T. L., and Jones, H. (2004). Whose job is it anyway? A study of human-robot interaction in a collaborative task. *Hum. Comput. Interact.* 19, 151–181. doi: 10.1207/s15327051hci1901&2_7

Hodgkin, A. (1952). A quantitative description of ion currents and its application and excitation in nerve membranes. *J. Physiol.* 117, 500–544. doi: 10.1113/jphysiol.1952.sp004764

Huang, G. B., Mattar, M., Berg, T., and Learned-Miller, E. (2008). Labeled faces in the wild: a database forstudying face recognition in unconstrained environments. HAL-Inria.

Ishi, C. T., Liu, C., Ishiguro, H., and Hagita, N. (2010). "Head motion during dialogue speech and nod timing control in humanoid robots," in *2010 5th ACM/IEEE International Conference on Human-Robot Interaction (HRI)* (Osaka: IEEE), 293–300. doi: 10.1109/HRI.2010.5453183

Kazemi, V., and Sullivan, J. (2014). "One millisecond face alignment with an ensemble of regression trees," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (Columbus, OH), 1867–1874. doi: 10.1109/CVPR.2014.241

Kiesler, S., and Hinds, P. (2004). Introduction to this special issue on human-robot interaction. *Hum. Comput. Interact.* 19, 1–8. doi: 10.1207/s15327051hci1901&2_1

King, D. E. (2009). DLIB-ML: a machine learning toolkit. *J. Mach. Learn. Res.* 10, 1755–1758.

King, D. E. (2015). Max-margin object detection. *arXiv preprint arXiv:1502.00046.*

Kousidis, S., and Schlangen, D. (2015). "The power of a glance: evaluating embodiment and turn-tracking strategies of an active robotic overhearer," in *2015 AAAI Spring Symposium Series* (Palo Alto, CA).

Kraft, D., and Schnepper, K. (1989). *SLSQP–A Nonlinear Programming Method With Quadratic Programming Subproblems*. DLR, Oberpfaffenhofen.

Lenc, L., and Král, P. (2015). Automatic face recognition system based on the sift features. *Comput. Electric. Eng.* 46, 256–272. doi: 10.1016/j.compeleceng.2015.01.014

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., et al. (2016). "SSD: single shot multibox detector," in *European Conference on Computer Vision* (Amsterdam: Springer), 21–37. doi: 10.1007/978-3-319-46448-0_2

Lowe, D. G. (1999). "Object recognition from local scale-invariant features," in *ICCV* (Kerkyra), Vol. 99, 1150–1157. doi: 10.1109/ICCV.1999.790410

Massé, B. (2018). *Gaze direction in the context of social human-robot interaction* (Ph.D. thesis).

Milborrow, S., and Nicolls, F. (2008). "Locating facial features with an extended active shape model," in *European Conference on Computer Vision* (Marseille: Springer), 504–513. doi: 10.1007/978-3-540-88693-8_37

Milighetti, G., Vallone, L., and De Luca, A. (2011). "Adaptive predictive gaze control of a redundant humanoid robot head," in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems* (San Francisco, CA: IEEE), 3192–3198. doi: 10.1109/IROS.2011.6094417

Potapova, E., Zillich, M., and Vincze, M. (2017). Survey of recent advances in 3D visual attention for robotics. *Int.*

J. Robot. Res. 36, 1159–1176. doi: 10.1177/0278364917726587

Quigley, M., Conley, K., Gerkey, B., Faust, J., Foote, T., Leibs, J., et al. (2009). "ROS: an open-source robot operating system," in *ICRA Workshop on Open Source Software*, Vol. 3 (Kobe), 5.

Ren, S., He, K., Girshick, R., and Sun, J. (2015). "Faster R-CNN: towards real-time object detection with region proposal networks," in *Advances in Neural Information Processing Systems*, 91–99.

Rosales, R., and Sclaroff, S. (1998). *Improved Tracking of Multiple Humans With Trajectory Prediction and Occlusion Modeling*. Technical report, Boston University Computer Science Department.

Saenko, K., Livescu, K., Siracusa, M., Wilson, K., Glass, J., and Darrell, T. (2005). "Visual speech recognition with loosely synchronized feature streams," in *Tenth IEEE International Conference on Computer Vision (ICCV'05)* (Beijing: IEEE), 1424–1431. doi: 10.1109/ICCV.2005.251

Saldien, J., Vanderborght, B., Goris, K., Van Damme, M., and Lefeber, D. (2014). A motion system for social and animated robots. *Int. J. Adv. Robot. Syst.* 11. doi: 10.5772/58402

Schulz, D., Burgard, W., Fox, D., and Cremers, A. B. (2003). People tracking with mobile robots using sample-based joint probabilistic data association filters. *Int. J. Robot. Res.* 22, 99–116. doi: 10.1177/0278364903022002002

Shiomi, M., Kanda, T., Miralles, N., Miyashita, T., Fasel, I., Movellan, J., et al. (2004). "Face-to-face interactive humanoid robot," in *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Sendai: IEEE), 1340–1346. doi: 10.1109/IROS.2004.1389582

Siatras, S., Nikolaidis, N., Krinidis, M., and Pitas, I. (2008). Visual lip activity detection and speaker detection using mouth region intensities. *IEEE Trans. Circ. Syst. Video Technol.* 19, 133–137. doi: 10.1109/TCSVT.2008.2009262

Sidner, C. L., Kidd, C. D., Lee, C., and Lesh, N. (2004). "Where to look: a study of human-robot engagement," in *Proceedings of the 9th International Conference on Intelligent User Interfaces* (Funchal: ACM), 78–84. doi: 10.1145/964442.964458

Thrun, S. (2004). Toward a framework for human-robot interaction. *Hum. Comput. Interact.* 19, 9–24. doi: 10.1207/s15327051hci1901&2_2

Vega, J., Perdices, E., and Cañas, J. (2013). Robot evolutionary localization based on attentive visual short-term memory. *Sensors* 13, 1268–1299. doi: 10.3390/s130101268

Viciana-Abad, R., Marfil, R., Perez-Lorenzo, J., Bandera, J., Romero-Garces, A., and Reche-Lopez, P. (2014). Audio-visual perception system for a humanoid robotic head. *Sensors* 14, 9522–9545. doi: 10.3390/s140609522

Viola, P., and Jones, M. (2001). "Rapid object detection using a boosted cascade of simple features," in *Computer Vision and Pattern Recognition 2001* (Kauai, HI), 511–518.

Viola, P., and Jones, M. J. (2004). Robust real-time face detection. *Int. J. Comput. Vis.* 57, 137–154. doi: 10.1023/B:VISI.0000013087.49260.fb

Zafeiriou, S., Zhang, C., and Zhang, Z. (2015). A survey on face detection in the wild: past, present and future. *Comput. Vis. Image Understand.* 138, 1–24. doi: 10.1016/j.cviu.2015.03.015

Zaraki, A., Mazzei, D., Giuliani, M., and De Rossi, D. (2014). Designing and evaluating a social gaze-control system for a humanoid robot. *IEEE Trans. Hum. Mach. Syst.* 44, 157–168. doi: 10.1109/THMS.2014.2303083

Data-Driven Optimal Assistance Control of a Lower Limb Exoskeleton for Hemiplegic Patients

Zhinan Peng[1], Rui Luo[1], Rui Huang[1]*, Tengbo Yu[2], Jiangping Hu[1], Kecheng Shi[1] and Hong Cheng[1]

[1] School of Automation Engineering, University of Electronic Science and Technology of China, Chengdu, China,
[2] Department of Sports Medicine, The Affiliated Hospital of Qingdao University, Qingdao, China

More recently, lower limb exoskeletons (LLE) have gained considerable interests in strength augmentation, rehabilitation, and walking assistance scenarios. For walking assistance, the LLE is expected to control the affected leg to track the unaffected leg's motion naturally. A critical issue in this scenario is that the exoskeleton system needs to deal with unpredictable disturbance from the patient, and the controller has the ability to adapt to different wearers. To this end, a novel data-driven optimal control (DDOC) strategy is proposed to adapt different hemiplegic patients with unpredictable disturbances. The interaction relation between two lower limbs of LLE and the leg of patient's unaffected side are modeled in the context of leader-follower framework. Then, the walking assistance control problem is transformed into an optimal control problem. A policy iteration (PI) algorithm is utilized to obtain the optimal controller. To improve the online adaptation to different patients, an actor-critic neural network (AC/NN) structure of the reinforcement learning (RL) is employed to learn the optimal controller on the basis of PI algorithm. Finally, experiments both on a simulation environment and a real LLE system are conducted to verify the effectiveness of the proposed walking assistance control method.

Keywords: walking assistance control, reinforcement learning, leader-follower multi-agent system, lower limb exoskeleton, hemiplegic patients, actor-critic neural network

1. INTRODUCTION

With the increasing requirement of accomplishing complex or difficult tasks in the fields of industry and human daily life, wearable devices/robots have attracted more attentions (Fang et al., 2018, 2019). As one of wearable devices, the lower limb exoskeleton (LLE) integrates artificial intelligence technologies, control and robotic theory, and has become a hot topic own to its practical applications. Note that so many injuries caused by neuromuscular diseases, and accidents reduce the walking ability, most assistive exoskeletons are developed to aid paraplegic patients (Dollar and Herr, 2008; Strausser and Kazerooni, 2011). On the other hand, stroke has gradually become a global health-care problem, which inspires many researchers to pay attention to walking assistance or rehabilitation case for hemiplegic individuals (Ho et al., 2011; Bortole et al., 2015; Iqbal and Baizid, 2015; Louie and Eng, 2016).

From the functions point of views, the LLE can be categorized in three classes of applications, namely, strength augmentation (Walsh et al., 2006; Huang et al., 2016, 2019), walking

assistance (Esquenazi et al., 2017; Zhang et al., 2017), and rehabilitation (Sankai, 2010; Huo et al., 2014). For strength augmentation, the wearers usually have walking ability, and the influence of human-robot interaction force should be considered in the controller designs. For walking assistance case, one usually uses the LLE to assist patients' walking/training in which the patients lose their ability to walk. Thus, LLE has served as a device for rehabilitation/walking training with paraplegia and hemiplegia. In recent, some researchers have introduced biological signals of human body into the controller designs, such as Electromyography signal (EMG) (Kiguchi et al., 2004) and Electroencephalogram signal (EEG) (Kilicarslan et al., 2013).

In the early research works of rehabilitation and gait recovery of hemiplegia, researcher proposed Ankle-Foot Orthosis (AFO) to achieve good recovery performance (Tyson and Thornton, 2001; Fatone et al., 2009). In order to provide active power assistance for hemiplegic patients, many powered orthosis with active motors have been developed, such as active AFO developed by Blaya and Herr (2004) and Series Elastic Remote Knee Actuator (SERKA) developed by Sulzer et al. (2009). However, these kinds of orthosis are designed for repairing local motion function of hemiplegic patients in particular scenarios, such as the SERKA is design for stroke patient with stiff-knee gait (SKG).

For the assistance control problem of LLE with hemiplegia, one usually focus on how to derive the LLE to generate a normal motion that aid the patients walking or recovering (Maciejasz et al., 2014; Hassan et al., 2018). Sankai developed a single leg exoskeleton system for hemiplegic patients based on the Hybrid Assistive Limb (HAL) (Kawamoto et al., 2009). For the studies on the HAL system with single leg case, motion information of the unaffected side is generated to synchronize gait of the affected side (Kawamoto et al., 2014). Note that the single leg based HAL system should be re-designed as the wearer has different disabled side. In Fisher et al. (2011), a powered exoskeleton was used to improve patients with hemiparesis walking function via robot assisted gait training. In Murray et al. (2014), the authors proposed a control approach of a LLE to provide walking assistance, without giving desired joint angle trajectory, for facilitating recovery. More recently, the walking assistance control problem for a LLE with hemiplegia was investigated via a learning-based control method (Huang et al., 2018).

In most of the existing relevant works, the case of disturbances caused by system or external environment has not been taken into consideration in the designs of controllers. In fact, disturbances caused by system or external environment will affect the control performance of system, which should be considered in controller designs. On the other hand, the precise system dynamics of exoskeleton is difficult to establish, which decreases the control performance of the *model-based* methods in real systems. To solve this issue, the system identification is needed that would introduce new approximation errors. Therefore, the motivation of this paper aims to address these problems.

Motivated by the above-mentioned discussions and observations. In this paper, a data-driven optimal control (DDOC) strategy is proposed for walking assistance of lower exoskeleton with hemiplegic patients. First, the interaction communications between the both two low limbs of LLE and

hemiplegic patient are modeled as a *leader-follower multi-agent system* (LFMAS) framework. Then, a policy iteration (PI) algorithm is employed to compute the optimal assistance controller. Further, in order to improve adaptive performance for walking assistance with different hemiplegic patients, a RL method, called actor-critic neural network (AC/NN), is proposed to achieve better control performance, where the learning process only relays on measurement data from the LLE system. The main contributions of this paper can be summarized as follows:

1. Different from most of the existing control method which is designed in a model-based fashion, a DDOC strategy based on PI algorithm is proposed to learn the optimal assistance controller for walking. The proposed method is designed in a model-free manner without the requirement of the complete knowledge about the accurate dynamics of the exoskeleton system and system identification.
2. An adaptive online-learning based AC/NN structure is employed for the implementation of the controller design, which aims to perform adaptability performance for different patients and achieve good robust against disturbances.

Moreover, the proposed DDOC method is validated through a two degree-of-freedom (2-DOF) simulation environment, and then it is successfully applied on a real LLE system with healthy subjects who simulate paraplegia. Both simulation and experimental results verify that the proposed control approach has robustness performance against disturbances and has adaptive ability for different wearers or even the same wearer with different gait patterns.

The rest of this paper is organized as follows. In section 2, the modeling process of LFMAS for exoskeleton system with hemiplegic patients is established, the system dynamics of the exoskeleton and problem formulation are given. Then, section 3 proposes the PI based optimal assistance controller designs. Section 4 proposes the data-driven adaptive control strategy by making using of RL framework on the basis of the PI algorithm. In section 5, the proposed control methods are illustrated in simulation scenario and is applied to an actual exoskeleton system with healthy people who simulate hemiplegic patients in section 6. Section 7 gives the conclusions and future work.

2. MODELING AND PROBLEM FORMULATION

In this section, the modeling process for the LLE with hemiplegic patients, namely LFMAS, is given to describe the interaction relations among both lower limbs of LLE and patients' legs. An information exchange rule is introduced for the LFMAS. Then, the system dynamics and control problem are formulated.

2.1. Modeling Exoskeleton System as LFMAS

In this paper, the focus is aim at designing an adaptive assistance controller of a LLE system with both lower extremities to assist hemiplegic individual walking. It should be noted that, for

FIGURE 1 | The modeling of Leader-Follower Multi-Agent System. **(A)** The schematic diagram of the LFMAS (The notations 0, 1, and 2 denote leader, follower 1, and follower 2, respectively). **(B)** Communication topology network of the LFMAS structure.
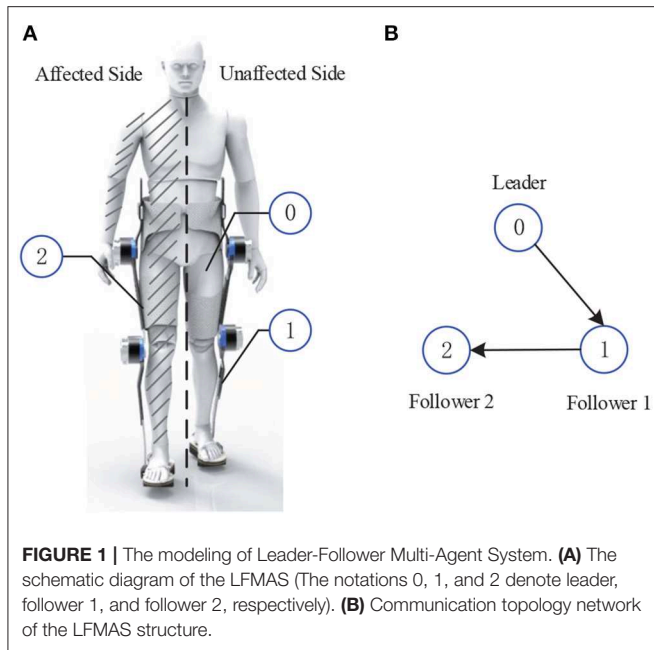
hemiplegic patients, one of the two legs usually loses walking ability and the other one is normal. Therefore, before introducing the controller designs, it is necessary to tackle how to model the interaction relations among them appropriately such that both low limbs and the LLE can achieve their mutual communication.

In light of the cooperative distributed control, *leader-follower* mechanism has been wildly utilized in multi-agent systems control (Hu and Feng, 2010), where the main idea is that information interactions among agents are achieved in a distributed way. In this paper, this mechanism is extended to model the unaffected leg of the hemiplegic patients and the both lower extremities of exoskeleton as a LFMAS, where the structure of the LFMAS for exoskeleton system with hemiplegic individuals is illustrated in **Figure 1A**. That is, the exoskeleton with hemiplegia is divided into three components: one leader agent and two follower agents. In other words, the unaffected leg of patient is regarded as the leader of LFMAS, equipped with an Inertial Measurement Unit (IMU) sensors for measuring its joints' states. Furthermore, both two lower extremities of the LLE system are defined as two follower agents, i.e., follower 1 and follower 2 which can be described as follows:

1. *Follower* 1 is the exoskeleton leg of unaffected side, which synchronizes the leader agent's (unaffected side of patient's leg) motion immediately.
2. *Follower* 2 is the other side of exoskeleton's limb with the disabled leg of patient, the patient's disabled leg is tightly connected with the exoskeleton.

In the framework of LFMAS, it should be pointed that there is a phase difference between the motion of the affected side and the unaffected side, naturally, In other words, follower 1 first synchronize to the leader's motion trajectory and then follower 2 is expected to track to the leader's trajectory motion after half gait cycle interval.

To guarantee walking assistance control performance, on the basis of LFMAS, the information interaction scheme should be designed for both lower extremities and patient's legs, which means that the information/data (LLE's state and control signal) can be transmitted among them. To this end, the following gives an information exchange rule to describe the evolution of the agents' communication.

(i) Information Evolution Rule: The information update for follower agent i ($i = 1, 2$) includes combining its own information with those received from its neighbors, and Leader can transmit its information to Follower. Assume that each agent has a weight vector $a_i = [a_{ij}]$, in which each element a_{ij} represents that agent i assigns to the information obtained from a neighboring agent j. **Figure 1B** denotes the communication topology network between agents where arrows indicate the direction of information flow.

(ii) Weight Rule: Let $\mathcal{N}(i)$ be the neighbors set of the *ith* Follower agent. For arbitrary $i \in \{1, 2\}$, if $j \in \mathcal{N}(i)$, $a_{ij} > 0$; if $j \notin \mathcal{N}(i)$, $a_{ij} = 0$. Let $\sum_{j \in \mathcal{N}(i)} a_{ij} = d_i$ be the sum of the neighbors' weights for agent i.

2.2. Dynamics Model of LLE System

In this paper, the dynamics of the LLE system is described as a second-order nonlinear mechanical system (i.e., Euler-lagrange system). Therefore, the dynamics of the both lower extremities, i.e., follower 1 ($i = 1$) and follower 2 ($i = 2$) of the exoskeleton are described as follows:

$$H_i(q_i)\ddot{q}_i + C_i(q_i, \dot{q}_i)\dot{q}_i + G_i(q_i) = \tau_i, \quad i = 1, 2 \qquad (1)$$

where $q_i = (q_{ih}, q_{ik})^\top \in R^2$ denotes the joints' angle of the LLE, q_{ih} and q_{ik} represent the hip joint and knee joint, respectively. $H_i(q_i)$ denotes inertia matrix, $C_i(q_i, \dot{q}_i)$ represents the centripetal and coriolis matrix. $G_i(q_i)$ denotes the gravitation term, $\tau_i = (\tau_{iu}, \tau_{id})^\top$ are the input torques generated by up and down motors for hip and knee joint. Further, we can rewrite Equation (1) as a state-space form:

$$\begin{bmatrix} \dot{q}_i \\ \ddot{q}_i \end{bmatrix} = \begin{bmatrix} \dot{q}_i \\ -H_i^{-1}(C_i\dot{q}_i + G_i) \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ H_i^{-1} \end{bmatrix} \tau_i$$

or equivalently,

$$\dot{\eta}_i(t) = f_i(\eta_i(t)) + g_i(\eta_i(t))u_i, \qquad (2)$$

where $\eta_i(t) = [q_i^\top, \dot{q}_i^\top]^\top$, $g_i(\eta_i(t)) = \begin{bmatrix} \mathbf{0} \\ H_i^{-1} \end{bmatrix}$, $f_i(\eta_i(t)) = \begin{bmatrix} \mathbf{0} & I \\ \mathbf{0} & -H_i^{-1}C_i \end{bmatrix} \eta_i(t) + \begin{bmatrix} \mathbf{0} \\ -H_i^{-1}G_i \end{bmatrix}$, $\tau_i = u_i$.

The dynamic of the leader (the motion trajectory of patient's unaffected leg) is given by:

$$\dot{\eta}_r(t) = f(\eta_r(t)), \qquad (3)$$

where $\eta_r(t)$ indicates the joint angle collected from human via an IMU sensors matched on the pilot's leg.

2.2.1. Design Objective

The goal is to generate the controller strategy u_i to ensure the trajectory $\eta_i(t)$ generated by Equation (2) can track the trajectory $\eta_r(t)$ in Equation (3). That is, it is desired to make the following tracking error index go to zero:

$$\lim_{t \to \infty} \|\eta_i(t) - \eta_r(t)\| = 0. \tag{4}$$

In order to achieve control objective, the local neighbor tracking errors of dynamics (2) for follower i are defined as

$$\xi_i(t) = \sum_{j \in \mathcal{N}(i)} a_{ij}(\eta_i(t) - \eta_j(t)) + c_i(\eta_i(t) - \eta_r(t)), \tag{5}$$

where $\mathcal{N}(i)$ and a_{ij} have been defined in section 2.1. $c_i > 0$ denotes the pinning gain, which means agent i can obtain the Leader's information.

Taking the derivation of Equation (5), combining Equation (2) and Equation (3), the dynamics of the tracking errors are written as

$$\dot{\xi}_i(t) = f_{\xi_i} + (d_i + c_i)g_i(t)u_i(t) - \sum_{j \in \mathcal{N}(i)} a_{ij}g_j(t)u_j(t), \tag{6}$$

where $f_{\xi_i}(t) = \sum_{j \in \mathcal{N}(i)} a_{ij}(f_i - f_j) + c_i(f_i - f)$, d_i indicates the sum of the weights of the ith follower agent.

3. POLICY ITERATION BASED CONTROLLER

Based on the system modeling and problem formulation, in this section, the walking assistance control problem will be transformed to an optimal control problem by introducing local cost functions and using optimization theories. Then, the state-of-the-art algorithm called policy iteration (PI) is proposed to obtain the solution to the coupled Hamilton-Jacobi-Bellman (HJB) equation, and thus the optimal controller $u_i^*(t)$ is obtained for solving walking assistance problem.

From the perspective of optimal control (Vamvoudakis and Lewis, 2010) and inspired by RL methods (Mnih et al., 2015, 2016; Sutton and Barto, 2018), we use a local cost function to assess the long-term learning and control performance, which is defined as follows:

$$V_i(\xi_i(t)) = \int_t^{\infty} r_i\big(\xi_i(s), u_i(s), u_{(j)}(s)\big)ds, \tag{7}$$

where $u_{(j)}(t)$ denotes the neighbors' control of Follower agent i, and $r_i\big(\xi_i(t), u_i(t), u_{(j)}(t)\big) = \xi_i^{\top}(t)Q_{ii}\xi_i(t) + u_i^{\top}(t)P_{ii}u_i(t) + \sum_{j \in \mathcal{N}(i)} u_i^{\top}(t)S_{ij}u_{(j)}(t)$ is the reward function, where the $Q_{ii} > 0$, $P_{ii} > 0$ and $S_{ij} > 0$ are symmetric positive definite weighting matrices, respectively. For the notation simplification, we set $r_i(\xi_i(t), u_i(t), u_{(j)}(t)) = r_i(\xi_i(t), u_i(t))$.

Till now, the walking assistance control problem is transformed into an optimal control problem, which aims to design a distributed controller to guarantee the *Design Objective* as well as minimizing the local cost function (Equation 7).

Further, the Hamilton function is represented as

$$H_i\big(\xi_i(t), u_i(t), V_i(\xi_i(t))\big) = r_i(\xi_i(t), u_i(t)) + \nabla V_{\xi}^{\top} \dot{\xi}_i(t), \tag{8}$$

where $V_i(0) = 0$, $\nabla V_{\xi} = \partial V_i(\xi_i(t))/\partial \xi_i(t)$ is a partial differential part.

Using the stationary condition for Equation (8), i.e., let $\partial H_i(t)/\partial u_i(t) = 0$, the optimal controller $u_i^*(t)$ is obtained as

$$u_i^*(t) = -\frac{1}{2}(d_i + c_i)P_{ii}^{-1}g_i^{\top}(t)\nabla V_{\xi}. \tag{9}$$

The optimal cost function $V_i^*(\xi_i(t))$ satisfies the following coupled Hamilton-Jacobi-Bellman (HJB) equation:

$$H_i\big(\xi_i(t), u_i^*(t), V_i^*(\xi(t))\big) = r_i(\xi_i(t), u_i^*(t)) + \nabla V_{\xi}^{*\top} \dot{\xi}_i(t) = 0. \tag{10}$$

Since the coupled HJB equation Equation (10) exists the nonlinear item and partial differential part, which makes it hard to be solved analytically. Therefore, the PI algorithm (Liu and Wei, 2014; Wang et al., 2014), is introduced to approximate the HJB equation and cope with this issue by a successive iteration way.

Let $u_i^l(t)$ and $V_i^l(\xi_i(t))$ represent iterative control and iterative Q-function, respectively, with l is iteration index. There are two components in PI algorithm, one is *policy evaluation* and the other is *policy improvement*. The detailed iterative performing process can be summarized as follows:

PI Algorithm: Start with admissible initial control u_i^0.

Step 1. *Policy Evaluation*: Given the control policy u_i^l, solve for value function $V_i^l(\xi(t))$ by

$$H_i\big(\xi_i(t), u_i^l(t), V_i^l(\xi(t))\big) = r_i(\xi_i(t), u_i^l(t)) + \nabla V_{\xi}^{l\top} \dot{\xi}_i(t) = 0. \tag{11}$$

Step 2. *Policy Improvement*: Compute the control law u_i^l by

$$u_i^{l+1}(t) = -\frac{1}{2}(d_i + c_i)P_{ii}^{-1}g_i^{\top}(t)\nabla V_{\xi}^l. \tag{12}$$

Step 3. If $\|V_i^l(\xi_i) - V_i^{l-1}(\xi_i)\| \le \epsilon$ (ϵ is a small positive constant), end. Else, let $l = l + 1$, go to step 1.

The PI algorithm is an effective method to solve the various optimal control problems. It has been proved that the iterative cost function and the iterative control strategy in PI will converge to the optimal values $V_i^{\star}(t)$ and $u_i^{\star}(t)$ through iterations (Peng et al., 2019, 2020).

It is worth noting from the above algorithm that the PI algorithm requires the knowledge of system models, i.e., $g_i(t)$ exists in the controller (Equation 12). In this sense, system identification is needed normally (Ghan and Kazerooni, 2006), but it is not suitable for the practical exoskeleton system with different hemiplegic patient. Since for different wearers/patients, the identification process needs to be reconstructed. To overcome this difficulty, the following section will present a data-driven adaptive control strategy with an online-learning fashion. It should be emphasized that this method avoids needing the knowledge of the accurate system dynamics, and no system identification is introduced.

4. IMPLEMENTATION OF CONTROLLER DESIGN

In this section, we will present the DDOC algorithm base on PI algorithm to achieve online-learning-based control and better adaptive performance for different patients via a neural network (NN) framework of RL called AC/NN. In the AC/NN, actor network is used to approximate controller and critic network is introduced to estimate cost function online, respectively. The detailed descriptions are given as follows.

4.1. The Critic NN Modular

First, the critic networks are adopted to approximate the cost function $V_i(t)$ as follows:

$$\hat{V}_i(t) = \hat{w}_{ci}^\top(t)\varphi_{ci}(z_{ci}(t)), \tag{13}$$

where z_{ci} is an input information of the critic modular and information from ξ_i, u_i, and $u_{(j)}$, $\varphi_{ci}(z_{ci})$ denotes the activation function, and $\hat{W}_{ci}$ is the weight vector of the critic network modular.

Then, at each time step, the Hamilton function (8) can be approximated as follows:

$$\mathfrak{e}_{ci}(t) = \int_t^{t+\triangle T} r_i(\xi_i, u_i)ds + \hat{V}_i(t+\triangle T) - \hat{V}_i(t)$$
$$= \int_t^{t+\triangle T} r_i(\xi_i, u_i)ds + \hat{w}_{ci}^\top(\varphi_{ci}(z_{ci}(t+\triangle T)) - \varphi_{ci}(z_{ci}(t))), \tag{14}$$

where $\triangle T > 0$ denotes the time interval.

Then, the Equation (14) is utilized to define the approximation error for the critic NNs. Thus, the squared residual error function to be minimized is defined as

$$E_{ci}(t) = \frac{1}{2}\|\mathfrak{e}_{ci}(t)\|^2 = \frac{1}{2}\mathfrak{e}_{ci}^2(t). \tag{15}$$

Then, by making use of gradient descent based weight update rule (Si and Wang, 2001), the tuning weight law can be adopted as follows

$$\dot{\hat{w}}_{ci}(t) = -\varrho_{ci}\frac{\partial E_{ci}(t)}{\partial \mathfrak{e}_{ci}(t)}\frac{\partial \mathfrak{e}_{ci}(t)}{\partial \hat{V}_i(t)}\frac{\partial \hat{V}_i(t)}{\partial \hat{w}_{ci}(t)}$$
$$= -\varrho_{ci}\varphi_{ci}(z_{ci})(\hat{w}_{ci}^\top \Delta\varphi_{ci}(z_{ci}) + \int_t^{t+\triangle T} r_i(\xi_i, u_i)ds), \tag{16}$$

where $\Delta\varphi_{ci}(z_{ci}) = \varphi_{ci}(z_{ci}(t+\triangle T)) - \varphi_{ci}(z_{ci}(t))$, ϱ_{ci} is the learning rate of the critic network modular for agent i.

4.2. The Actor NN Modular

Next, define the actor neural networks, which is employed to estimate the control strategy, as follows:

$$\hat{u}_i(t) = \hat{w}_{ai}^\top(t)\varphi_{ai}(z_{ai}(t)), \tag{17}$$

where z_{ai} is an input vector of the actor network including ξ_i of agent i, $\varphi_{ai}(z_{ai})$ denotes the activation function, and $\hat{w}_{ai}$ is the weight matrix.

Algorithm 1: Optimal Walking Assistance Control Algorithm.

1: **Initialization**
2: Initialize the values of critic weight $\hat{w}_{ci}(0)$ and actor weight $\hat{w}_{ai}(0)$;
3: Set the learning rates of the critic network and actor network to be ρ_{ai} and ρ_{ci};
4: Choose a sufficiently small computation precision ϵ;
5: Let Q_{ii}, P_{ii} and S_{ij} be positive definite weighting matrices;
6: **repeat**
7:     Calculate the actor network to estimate the control strategy $\hat{u}_i \leftarrow$ (17);
8:     Calculate the critic network to estimate the cost function $\hat{V}_i \leftarrow$ (13);
9:     According to the available system data q_i and q_r, compute the error $\xi_i \leftarrow$ (5);
10:     Calculate the objective function E_{ci};
11:     Update the weights in the critic NNs using $\hat{w}_{ci}(t) \leftarrow$ (16);
12:     Calculate the objective function E_{ai};
13:     Update the weights in the actor NNs using $\hat{w}_{ai}(t) \leftarrow$ (20);
14: **until** $\|\hat{w}_{ci}' - \hat{w}_{ci}\| \leq \epsilon$ ($\hat{w}_{ci}$ and $\hat{w}_{ci}'$ denote the weight of the current time and previous time);

Then, in order to obtain the desired approximation optimal controller to minimize the cost function $\hat{V}_i$, the error function of the actor network is defined as

$$\mathfrak{e}_{ai}(t) = \hat{V}_i(\xi_i(t)) - \mathcal{U}_{obj}, \tag{18}$$

where $\mathcal{U}_{obj}$ is the ultimate objective function. From perspective of the RL, the value of the $\mathcal{U}_{obj}$ is selected according to different purposes of applications.

The squared residual error function to be minimized in the actor network is given by

$$E_{ai}(t) = \frac{1}{2}\|\mathfrak{e}_{ci}(t)\|^2 = \frac{1}{2}\mathfrak{e}_{ai}^2(t). \tag{19}$$

Similarly, with the aid of the gradient descent rule, the following updating rule for the actor network is obtained

$$\dot{\hat{w}}_{ai}(t) = -\varrho_{ai}\frac{\partial E_{ai}(t)}{\partial \mathfrak{e}_{ai}(t)}\frac{\partial \mathfrak{e}_{ai}(t)}{\partial \hat{V}_i(t)}\frac{\partial \hat{V}_i(t)}{\partial z_{ci}(t)}\frac{\partial z_{ci}(t)}{\partial \hat{u}_i(t)}\frac{\partial \hat{u}_i(t)}{\partial \hat{w}_{ai}(t)}$$
$$= -\varrho_{ai}\varphi_{ai}(z_{ai})\hat{w}_{ci}^\top \nabla\varphi_{ci}(z_{ci})\xi_i\varphi_{ci}^\top(z_{ci})\hat{w}_{ci}, \tag{20}$$

where $\xi_i = \partial z_{ci}/\partial \hat{u}_i$, $\nabla\varphi_{ci}(z_{ci}) = \partial\varphi_{ci}(z_{ci})/\partial z_{ci}$ and ϱ_{ai} is a learning rate of the actor NN for agent i.

The procedure of the data-driven adaptive control strategy is presented in Algorithm 1. It should be noted that only the measured system data, i.e., ξ_i and u_i are required in the design of the DDOC algorithm. Thus, this method is a data-driven/model-free approach, which improves the potential application of the proposed control method in real systems.

It is noted that in the neural network based approximated structure, some common forms of the activation functions are polynomial functions, tanh functions, sigmoid functions, and so

on. Further, we found that the appropriate selection of activation function is very important, which leads to the different size of the NN weight parameters. In this paper, the selection of activation function has the same dimension as the input data. These settings can reduce a huge computation burden for implementation.

5. NUMERICAL SIMULATION

In this section, a *2-DOF manipulator* system in simulation scenario is first carried out to validate the effectiveness of the proposed data-driven control strategy.

5.1. Dynamic Model of 2-DOF System

For simulation, the simulation environment is set up in Simulink-Matlab. The dynamics of the two followers are the same as Equation (1), where the system matrices are given as follows: $H_i = \begin{bmatrix} m_{i1} + m_{i2} + 2m_{i3}\cos(q_{i2}) & m_{i2} + m_{i3}\cos(q_{i2}) \\ m_{i2} + m_{i3}\cos(q_{i2}) & m_{i2} \end{bmatrix}$, $C_i = \begin{bmatrix} -m_{i3}\dot{q}_{i2}\sin(q_{i2}) & -m_{i3}(\dot{q}_{i1}) + q_{i2}\sin(q_{i2}) \\ -m_{i3}\dot{q}_{i1} & 0 \end{bmatrix}$, and the $G_i = \begin{bmatrix} m_{i4}g\cos(q_{i1}) + m_{i5}g\cos(q_{i1} + q_{i2}) \\ m_{i5}g\cos(q_{i1} + q_{i2}) \end{bmatrix}$, $\tau_i = [\tau_{i1}, \tau_{i2}]^\top$, m_{ip} ($p = 1, 2, 3, 4, 5$) are the masses. Note that, in simulation case, the given dynamic system can be used to product system data needed in DDOC algorithm.

The leader system (desired trajectories) is expressed by

$$q_r = \begin{bmatrix} q_{1r} \\ q_{2r} \end{bmatrix} = \begin{bmatrix} 0.5cos(t) + 0.2sin(3t) \\ 0.3cos(3t) - 0.5sin(2t) \end{bmatrix}. \tag{21}$$

We select the structure of the AC/NN as 3-layers back propagation (BP) NN (Goh, 1995). The initial values of critic NN weights and actor NN weights are set to be zero, and setting the value of the computation precision as $\epsilon = 10^{-5}$. The weight learning rates of the actor network and the critic network are chosen as $\rho_{ai} = 0.03$, $\rho_{ci} = 0.06$. The activation functions φ_{ai} and φ_{ci} are selected as the hyperbolic tangent functions, i.e., $\tanh(x) = (e^x - e^{-x})/(e^x + e^{-x})$.

5.2. Simulation Results and Analysis

As shown in **Figure 2**, we can see that after 2 s learning process, the critic NN weights and the actor NN weights are convergent, and thus the optimal weights parameters are obtained. Therefore, the approximate optimal controller can be obtained in Equation (17). On the basis of the optimal controller, the trajectory of joint angles $q_1 = (q_{11}, q_{12})^\top$ of follower 1 achieves a good tracking performance to the leader at $3 \text{ s} < t < 6$ s, which is illustrated in **Figure 3**.

In order to further verify the performance of adaption against uncertainty, we add some disturbance signal (white noise) to the system at $t \in [6, 7]$ s. In **Figure 2**, the AC/NN weights are retrained for learning again adaptively until converge from $t = 6$ s to $t = 8$ s, and thus the optimal controller has been modified correspondingly. With the help of the modified optimal controller, from **Figure 3**, it is seen that joint angle trajectories of two links of follower 1 are synchronized with the leader again quickly after $t = 8$ s. These simulation results illustrate the
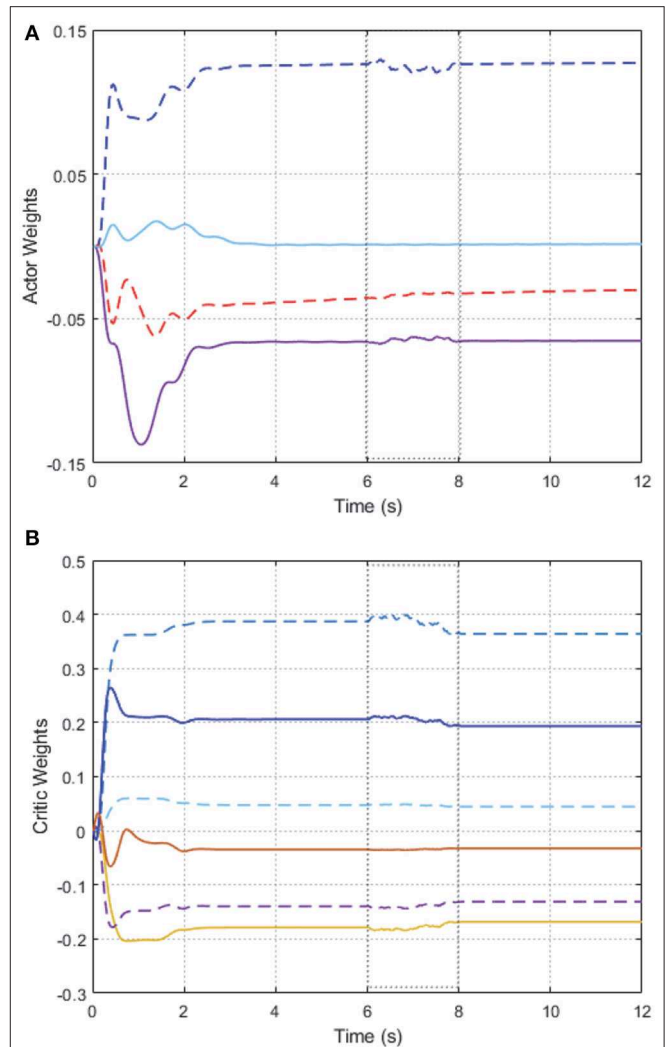
FIGURE 2 | Convergence of the AC/NN weights on 2-DOF simulation platform. **(A)** Actor network. **(B)** Critic network.

better control performance of the proposed DDOC algorithm, which has ability to respond to disturbances online in the system operation. It is proved that our proposed control method has good robustness against uncertainties.

6. EXPERIMENTS ON A REAL LLE SYSTEM

In this section, to further verify the control performance of the proposed data-driven control strategy, *walking assistance* experiments on an actual LLE system are performed.

6.1. Experimental Setup

To demonstrate the effectiveness and adaptability of the proposed control strategy, a practical LLE system, called AIDER, which is shown in **Figure 4**, is designed for walking assistance case to help hemiplegia. A distributed control system is embedded in AIDER

FIGURE 3 | The trajectories tracking performance of joint angle of follower 1 on 2-DOF simulation platform. **(A)** Link 1. **(B)** Link 2.

FIGURE 4 | The LLE system called AIDER for hemiplegic patient. 1. The subject/wearer; 2. Smart shoes with plantar pressure sensors inside; 3. The load backpack with embedded computer, IMU and power unit; 4. Active joints with node controllers (hip joints and knee joints).

which consists of a main controller and four node controllers. The main controller is placed on the backpack to compute the control algorithm. Node controllers are fixed near by the corresponding active joints position of LLE robot which aims to receive sensor data and execute control commands according to the main controller.

During the experiments on the AIDER system, three healthy subjects (1, 2, 3) with different heights (165, 176, 180 cm) are selected to participate this experiment and operate the wearable LLE robot. All wearers are simulated as hemiplegic patients, and the right legs of the subjects are simulated as the affected leg. In the walking assistance task for all wearers, each wearer is asked to walk for 50 s via the AIDER system. All the pilot's walking speed is varying from 0.1 to $0.4 m/s$. Further, the AIDER is equipped with accelerometer and the wearable sensory system for measuring system data.
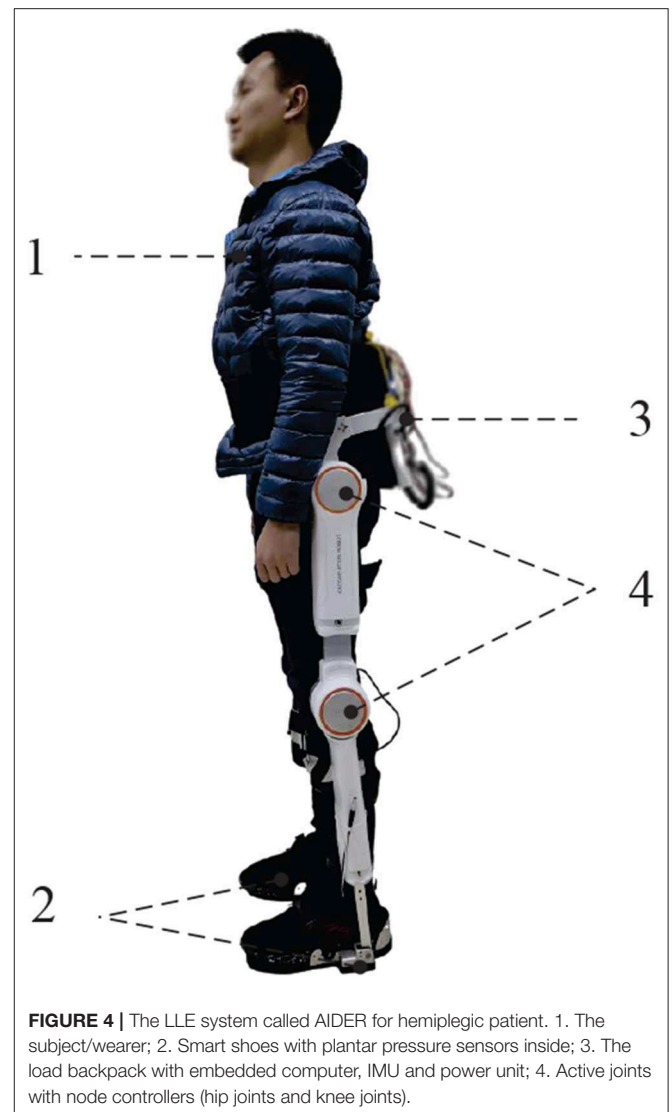
For the implementation of the proposed data-driven control strategy on the AIDER system. Note that the proposed data-driven control strategy DDOC has a learning process using the online system data at the beginning, which aims to adapt different subjects. After the learning stage, the optimal control policies can be obtained, and then walking assistance can be realized for the LLE system with pilots. We choose the AC/NN as 3-layer Back propagation (BP) NNs structure (Goh, 1995), that is, input layer, hidden layer and output layer. The initial values of weights $\hat{w}_{ci}$ and $\hat{w}_{ai}$ of the critic and actor are all set to be zero, and the activation functions φ_{ai} and φ_{ci} are chosen as hyperbolic tangent functions $\tanh(x) = (e^x - e^{-x})/(e^x + e^{-x})$. The learning rates are the same as in the simulation, that is $\rho_{ai} = 0.03$, $\rho_{ci} = 0.06$.

6.2. Experimental Results and Discussions

For participant 1, from **Figures 5A,B**, we can see that, after about 5 s training, the weights of AC/NN are bounded convergent, i.e., uniformly ultimately bounded because of the disturbances

FIGURE 5 | The trajectories of the AC/NN weights for AIDER with subject 1 in the experiment: **(A)** Actor weights. **(B)** Critic weights.

FIGURE 6 | The tracking control performance performance of the proposed DDOC strategy on AIDER with subject 1 in the experiment: **(A)** Hip joint's angle. **(B)** Knee joint's angle.
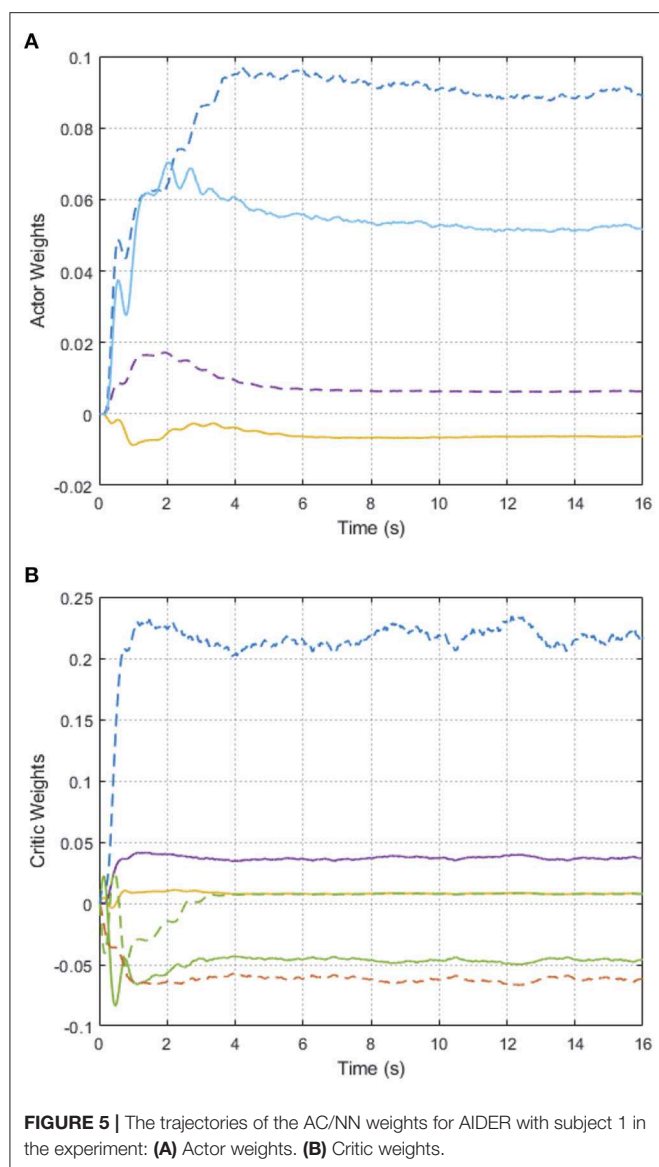
and uncertainties always exist in LLE system. The tracking performance of the hip joint and knee joint for the LLE system with wearer 1 is depicted in **Figures 6A,B**, which states that with the help of the learned optimal control policies, the hip joint and knee joint of two limbs of the exoskeleton (follower 1, 2) can achieve synchronization with the desired (leader's) motion trajectories. Moreover, it is noted that there are different walking motion patterns in the procedure of walking, which means our proposed method has capability of adapting different gait patterns. It should be pointed out that the affected side of wearer with exoskeleton's side (follower 2) has a half gait cycle delay to the side which has walking ability (leader), which is marked with blue dashed line as shown in **Figure 6**. In summary, the experimental results illustrate the effectiveness of the proposed DDOC approach for walking assistance of the exoskeleton with different pilots.
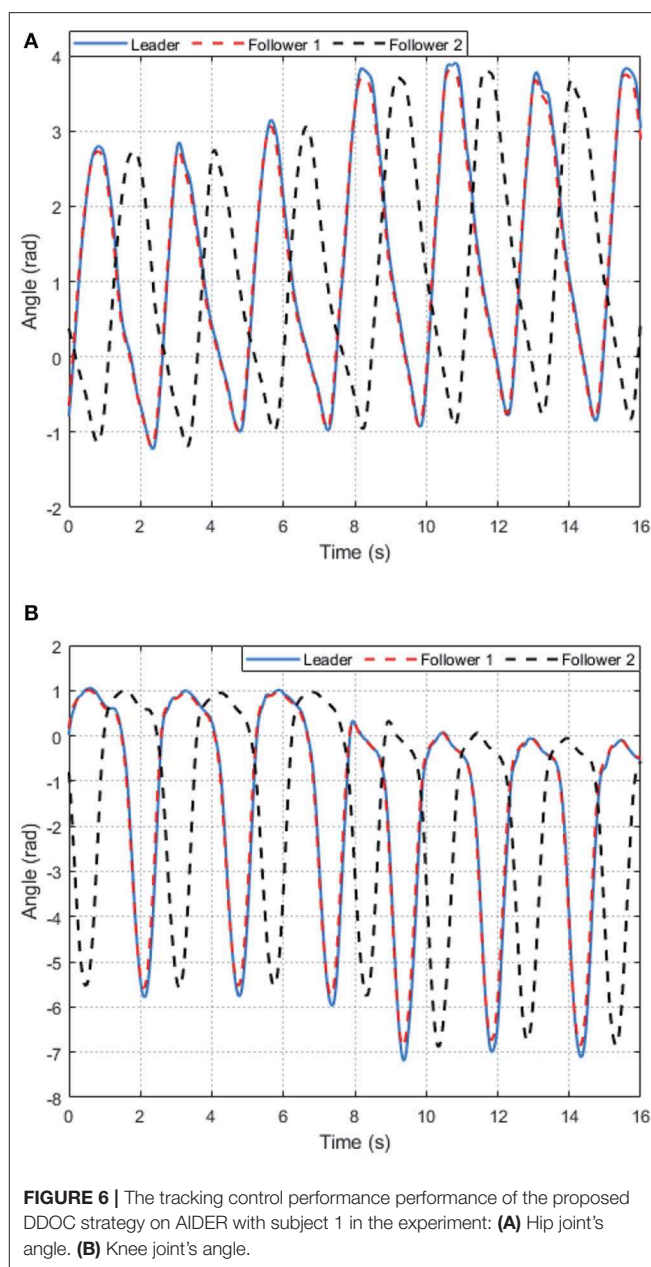
7. CONCLUSIONS

In this paper, a DDOC control strategy has been proposed for a lower exoskeleton system to assist hemiplegic patient walking. A LFMAS structure has been established to model the interaction relation among LLE system and hemiplegic individual. The walking assistance problem has been transformed to an optimal control problem. The PI algorithm has been introduced to obtain optimal assistance controller. On the basis of the PI algorithm, in order to adapt different patients, the AC/NN framework has been presented for the implementation of the proposed approach in an online-learning manner. It highlights that the controller design only relays on the measured system data, rather than the

accurate system model. Finally, we have successfully validated the proposed method on two situations: *2-DOF manipulator* in simulation environment and *walking assistance* experiment on a real LLE system called AIDER. Experimental results have confirmed the effectiveness of the proposed control method. In the future, we will focus on more practical control issues, and consider the RL-based controller designs for exoskeleton system with actuator faults and input time-delay.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/supplementary materials, further inquiries can be directed to the corresponding author/s.

ETHICS STATEMENT

All procedures performed in studies were approved by Research Ethical Committee of University of Electronic Science and Technology of China. Written informed consent

was obtained from the participant for the publication of any potentially identifiable images or data included in this article.

AUTHOR CONTRIBUTIONS

ZP designed the control methods, performed the experiments, and drafted the manuscript. RL, RH, and KS participated in the design of the controllers and assisted in the manuscript writing. JH and HC guided writing paper and doing experiments. TY designed procedure of experiments and helped to revise and improve the paper significantly.

REFERENCES

Blaya, J. A., and Herr, H. (2004). Adaptive control of a variable-impedance ankle-foot orthosis to assist drop-foot gait. *IEEE Trans. Neural Syst. Rehabil. Eng.* 12, 24–31. doi: 10.1109/TNSRE.2003.823266

Bortole, M., Venkatakrishnan, A., Zhu, F., Moreno, J. C., Francisco, G. E., Pons, J. L., et al. (2015). The H2 robotic exoskeleton for gait rehabilitation after stroke: early findings from a clinical study. *J. NeuroEng. Rehabil.* 12:54. doi: 10.1186/s12984-015-0048-y

Dollar, A. M., and Herr, H. (2008). Lower extremity exoskeletons and active orthoses: challenges and state-of-the-art. *IEEE Trans. Robot.* 24, 144–158. doi: 10.1109/TRO.2008.915453

Esquenazi, A., Talaty, M., and Jayaraman, A. (2017). Powered exoskeletons for walking assistance in persons with central nervous system injuries: a narrative review. *PM&R* 9, 46–62. doi: 10.1016/j.pmrj.2016.07.534

Fang, B., Sun, F., Liu, H., and Liu, C. (2018). 3D human gesture capturing and recognition by the immu-based data glove. *Neurocomputing* 277, 198–207. doi: 10.1016/j.neucom.2017.02.101

Fang, B., Wei, X., Sun, F., Huang, H., Yu, Y., and Liu, H. (2019). Skill learning for human-robot interaction using wearable device. *Tsinghua Sci. Technol.* 24, 654–662. doi: 10.26599/TST.2018.9010096

Fatone, S., Gard, S. A., and Malas, B. S. (2009). Effect of ankle-foot orthosis alignment and foot-plate length on the gait of adults with poststroke hemiplegia. *Arch. Phys. Med. Rehabil.* 90, 810–818. doi: 10.1016/j.apmr.2008.11.012

Fisher, S., Lucas, L., and Adam Thrasher, T. (2011). Robot-assisted gait training for patients with hemiparesis due to stroke. *Top. Stroke Rehabil.* 18, 269–276. doi: 10.1310/tsr1803-269

Ghan, J., and Kazerooni, H. (2006). "System identification for the Berkeley lower extremity exoskeleton (bleex)," in *IEEE International Conference on Robotics and Automation* (Orlando, FL), 3477–3484. doi: 10.1109/ROBOT.2006.1642233

Goh, A. T. (1995). Back-propagation neural networks for modeling complex systems. *Artif. Intell. Eng.* 9, 143–151. doi: 10.1016/0954-1810(94)00011-S

Hassan, M., Kadone, H., Ueno, T., Hada, Y., Sankai, Y., and Suzuki, K. (2018). Feasibility of synergy-based exoskeleton robot control in hemiplegia. *IEEE Trans. Neural Syst. Rehabil. Eng.* 26, 1233–1242. doi: 10.1109/TNSRE.2018.2832657

Ho, N., Tong, K., Hu, X., Fung, K., Wei, X., Rong, W., et al. (2011). "An EMG-driven exoskeleton hand robotic training device on chronic stroke subjects: task

training system for stroke rehabilitation," in *IEEE International Conference on Rehabilitation Robotics* (Zurich), 1–5. doi: 10.1109/ICORR.2011.5975340

Hu, J., and Feng, G. (2010). Distributed tracking control of leader-follower multi-agent systems under noisy measurement. *Automatica* 46, 1382–1387. doi: 10.1016/j.automatica.2010.05.020

Huang, R., Cheng, H., Guo, H., Chen, Q., and Lin, X. (2016). "Hierarchical interactive learning for a human-powered augmentation lower exoskeleton," in *IEEE International Conference on Robotics and Automation* (Stockholm), 257–263. doi: 10.1109/ICRA.2016.7487142

Huang, R., Cheng, H., Qiu, J., and Zhang, J. (2019). Learning physical human-robot interaction with coupled cooperative primitives for a lower exoskeleton. *IEEE Trans. Autom. Sci. Eng.* 16, 1566–1574. doi: 10.1109/TASE.2018.2886376

Huang, R., Peng, Z., Cheng, H., Hu, J., Qiu, J., Zou, C., et al. (2018). "Learning-based walking assistance control strategy for a lower limb exoskeleton with hemiplegia patients," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Madrid), 2280–2285. doi: 10.1109/IROS.2018.8594464

Huo, W., Mohammed, S., Moreno, J. C., and Amirat, Y. (2014). Lower limb wearable robots for assistance and rehabilitation: a state of the art. *IEEE Syst. J.* 10, 1068–1081. doi: 10.1109/JSYST.2014.2351491

Iqbal, J., and Baizid, K. (2015). Stroke rehabilitation using exoskeleton-based robotic exercisers: mini review. *Biomed. Res.* 26, 197–201.

Kawamoto, H., Hayashi, T., Sakurai, T., Eguchi, K., and Sankai, Y. (2009). "Development of single leg version of HAL for hemiplegia," in *Annual International Conference of the IEEE Engineering in Medicine and Biology Society* (Minneapolis, MN), 5038–5043. doi: 10.1109/IEMBS.2009.5333698

Kawamoto, H., Kandone, H., Sakurai, T., Ariyasu, R., Ueno, Y., Eguchi, K., et al. (2014). "Development of an assist controller with robot suit HAL for hemiplegic patients using motion data on the unaffected side," in *36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society* (Chicago, IL), 3077–3080. doi: 10.1109/EMBC.2014.6944273

Kiguchi, K., Tanaka, T., and Fukuda, T. (2004). Neuro-fuzzy control of a robotic exoskeleton with EMG signals. *IEEE Trans. Fuzzy Syst.* 12, 481–490. doi: 10.1109/TFUZZ.2004.832525

Kilicarslan, A., Prasad, S., Grossman, R. G., and Contreras-Vidal, J. L. (2013). "High accuracy decoding of user intentions using EEG to control a lower-body exoskeleton," in *35th annual International Conference of the IEEE Engineering in Medicine and Biology Society* (Osaka), 5606–5609. doi: 10.1109/EMBC.2013.6610821

Liu, D., and Wei, Q. (2014). Policy iteration adaptive dynamic programming algorithm for discrete-time nonlinear systems. *IEEE Trans. Neural Netw. Learn. Syst.* 25, 621–634. doi: 10.1109/TNNLS.2013.2281663

Louie, D. R., and Eng, J. J. (2016). Powered robotic exoskeletons in post-stroke rehabilitation of gait: a scoping review. *J. NeuroEng. Rehabil.* 13:53. doi: 10.1186/s12984-016-0162-5

Maciejasz, P., Eschweiler, J., Gerlach-Hahn, K., Jansen-Troy, A., and Leonhardt, S. (2014). A survey on robotic devices for upper limb rehabilitation. *J. NeuroEng. Rehabil.* 11:3. doi: 10.1186/1743-0003-11-3

Mnih, V., Badia, A. P., Mirza, M., Graves, A., Lillicrap, T., Harley, T., et al. (2016). "Asynchronous methods for deep reinforcement learning," in *International Conference on Machine Learning* (New York, NY), 1928–1937.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., et al. (2015). Human-level control through deep reinforcement learning. *Nature* 518, 529–533. doi: 10.1038/nature14236

Murray, S. A., Ha, K. H., Hartigan, C., and Goldfarb, M. (2014). An assistive control approach for a lower-limb exoskeleton to facilitate recovery of walking following stroke. *IEEE Trans. Neural Syst. Rehabil. Eng.* 23, 441–449. doi: 10.1109/TNSRE.2014.2346193

Peng, Z., Hu, J., Shi, K., Luo, R., Huang, R., Ghosh, B. K., et al. (2020). A novel optimal bipartite consensus control scheme for unknown multi-agent systems via model-free reinforcement learning. *Appl. Math. Comput.* 369:124821. doi: 10.1016/j.amc.2019.124821

Peng, Z., Zhao, Y., Hu, J., and Ghosh, B. K. (2019). Data-driven optimal tracking control of discrete-time multi-agent systems with two-stage policy iteration algorithm. *Inf. Sci.* 481, 189–202. doi: 10.1016/j.ins.2018.12.079

Sankai, Y. (2010). Hal: Hybrid assistive limb based on cybernics. *Robot. Res.* 66, 25–34. doi: 10.1007/978-3-642-14743-2_3

Si, J., and Wang, Y.-T. (2001). Online learning control by association and reinforcement. *IEEE Trans. Neural Netw.* 12, 264–276. doi: 10.1109/72.914523

Strausser, K. A., and Kazerooni, H. (2011). "The development and testing of a human machine interface for a mobile medical exoskeleton," in *IEEE/RSJ International Conference on Intelligent Robots and Systems* (San Francisco, CA), 4911–4916. doi: 10.1109/IROS.2011.6095025

Sulzer, J. S., Roiz, R. A., Peshkin, M. A., and Patton, J. L. (2009). A highly backdrivable, lightweight knee actuator for investigating gait in stroke. *IEEE Trans. Robot.* 25, 539–548. doi: 10.1109/TRO.2009.2019788

Sutton, R. S., and Barto, A. G. (2018). *Reinforcement Learning: An Introduction.* MIT Press.

Tyson, S., and Thornton, H. (2001). The effect of a hinged ankle foot orthosis on hemiplegic gait: objective measures and users' opinions. *Clin. Rehabil.* 15, 53–58. doi: 10.1191/026921501673858908

Vamvoudakis, K. G., and Lewis, F. L. (2010). Online actor-critic algorithm to solve the continuous-time infinite horizon optimal control problem. *Automatica* 46, 878–888. doi: 10.1016/j.automatica.2010.02.018

Walsh, C., Paluska, D., Pasch, K., Grand, W., Valiente, A., and Herr, H. (2006). "Development of a lightweight, under-actuated exoskeleton for load-carrying augmentation," in *IEEE International Conference on Robotics and Automation* (Orlando, FL), 3485–3491.

Wang, D., Liu, D., and Li, H. (2014). Policy iteration algorithm for online design of robust control for a class of continuous-time nonlinear systems. *IEEE Trans. Autom. Sci. Eng.* 11, 627–632. doi: 10.1109/TASE.2013.2296206

Zhang, J., Fiers, P., Witte, K. A., Jackson, R. W., Poggensee, K. L., Atkeson, C. G., et al. (2017). Human-in-the-loop optimization of exoskeleton assistance during walking. *Science* 356, 1280–1284. doi: 10.1126/science.aal5054

Check for
updates

An Incremental Learning Framework to Enhance Teaching by Demonstration Based on Multimodal Sensor Fusion

Jie Li[1], Junpei Zhong[2], Jingfeng Yang[3] and Chenguang Yang[4]*

[1] Key Laboratory of Autonomous Systems and Networked Control, School of Automation Science and Engineering, South China University of Technology, Guangzhou, China, [2] School of Science and Technology, Nottingham Trent University, Nottingham, United Kingdom, [3] Shenyang Institute of Automation Guangzhou Chinese Academy of Sciences, Guangzhou, China, [4] Bristol Robotics Laboratory, University of the West of England, Bristol, United Kingdom

Though a robot can reproduce the demonstration trajectory from a human demonstrator by teleoperation, there is a certain error between the reproduced trajectory and the desired trajectory. To minimize this error, we propose a multimodal incremental learning framework based on a teleoperation strategy that can enable the robot to reproduce the demonstration task accurately. The multimodal demonstration data are collected from two different kinds of sensors in the demonstration phase. Then, the Kalman filter (KF) and dynamic time warping (DTW) algorithms are used to preprocessing the data for the multiple sensor signals. The KF algorithm is mainly used to fuse sensor data of different modalities, and the DTW algorithm is used to align the data in the same timeline. The preprocessed demonstration data are further trained and learned by the incremental learning network and sent to a Baxter robot for reproducing the task demonstrated by the human. Comparative experiments have been performed to verify the effectiveness of the proposed framework.

Keywords: incremental learning network, teaching by demonstration, teleoperation, data fusion, robot learning

INTRODUCTION

With the development of control theory and sensor technology, robots have been widely applied in various fields, especially in industry and social service. It plays an increasingly vital role in human daily life, such as entertainment, education, and home service, etc. In most cases (Billard et al., 2008; Yang et al., 2018; Fang et al., 2019), robots need to learn and execute many complex and repetitive tasks, which include learning the motion skills from observing humans performing these tasks, also known as teaching by demonstration (TbD). TbD is an efficient approach to reduce the complexity of teaching a robot to perform new tasks (Billard et al., 2008; Yang et al., 2018). With this approach, a human tutor demonstrates how to implement a task to a robot easily (Ewerton et al., 2019). Then, the robot learns the key features from human demonstration and repeats it by itself. Obviously, the main issue of robot learning is how to learn more critical features from the demonstration to fulfill a certain task well. Therefore, it is essential to take account of some learning methods to learn much more useful features effectively. In this sense, robot learning contains two tasks: motion perception based on multiple sensors and features learning with efficient methods. Different modalities of sensors can enable obtaining an accurate description of the target motions

and enrich the information (Chavez-Garcia and Aycard, 2016), and the learning methods promote to learn the desirable features. In this paper, we developed a novel robot learning framework to enhance the performance of TbD. This framework combines the superiority of the incremental learning network and the multiple sensors fusion.

Multimodal sensor fusion is a promising technique to create more accurate, more complete, or more dependable data with less uncertainty (Elmenreich, 2002; Haghighat et al., 2011) to enrich the features of demonstration data. Compared with individual sensor data, the multi-sensor fusion data have a distinctive preponderance in four general aspects (Mitchell, 2007). First, a higher resolution and richer semantic become possible for usage in the representation of the data. Second, the fused sensory data or data from disparate sources can reduce the uncertainty of information than these sources are used individually. Besides, a more all-sided view regarding the object is allowed for a coherent space to enhance the completeness of the information. Last, if the data are noisy or have errors, the fusion process will reduce or eliminate noise and errors. Hence, the data through the fusion process are possible to achieve the desired result with enhanced reliability, extended parameter coverage, and improved resolution (Fung et al., 2017). The system with multiple sensors provides immense opportunities for applications in a wide variety of areas. Applications that benefit from the sensor fusion technology cover many engineering fields which include internet of things (Din et al., 2015; Bijarbooneh et al., 2016), automation systems (Iyengar et al., 2003; Caterina et al., 2015), computer vision (Eitel et al., 2015), target tracking (Smith and Singh, 2006), health care (Medjahed et al., 2011; Koshmak et al., 2016), mechatronics (Luo and Chang, 2012), and robotics (Chung et al., 2011).

Recently, the multimodal sensor fusion is widely engaged in human–robot interaction (HRI) to enhance the performance of interaction (Gui et al., 2017; Argyrou et al., 2018; Deng et al., 2018; Fang et al., 2019; Li C. et al., 2019). Gui et al. (2017) designed a multimodal rehabilitation HRI system, which combines the electroencephalogram (EEG)-based HRI and electromyography (EMG)-based HRI to assistant gait pattern, to enhance active participation of users for gait rehabilitation and to accomplish abundant locomotion modes for the exoskeleton. Argyrou et al. (2018) proposed a human–robot collaborative monitoring system that can fuse data from multiple sources to estimate the execution status of the tasks more accurately. Deng et al. (2018) proposed an improved HRI by fusing the operator's gesture and speech to control the movements of a robot. The fusion of gesture and speech improved the accuracy, efficiency, and naturalness of the proposed system. Li C. et al. (2019) developed an augmented reality interface based on HRI that the Kalman filter (KF) algorithm was used to fuse the position and velocity signals from the Leap Motion sensor and the Kinect sensor to improve the tracking performance, aiming to provide an easier and accurate interaction. Wan et al. (2017) developed an intelligent system to teach robots to do object assembly through multimodal vision for next-generation industrial assembly. Zeng et al. (2019) proposed a TbD system to teach the robots to learn specific tasks based on multiple sensor

fusion. Compared with single modal data, the multimodal data provide a more rich and complementary information source to facilitate the diversity of robot TbD. These applications benefit from sensor fusion technology because of multi-sensor-based data fusion algorithms. Due to the varieties of the nature of the fusion process, different algorithms are used to enable the different levels of sensor fusion, such as KF (Kalman, 1960), support vector machine (SVM) (Cortes and Vapnik, 1995; Waske and Benediktsson, 2007), particle filter (Crisan and Doucet, 2002), Bayesian inference method (Khaleghi et al., 2013), fuzzy sensor fusion approach (Gibson et al., 1994), and artificial neural network (Hu, 2010), etc. Studies showed that the KF is ideally suited to coping with multi-sensor estimation and data fusion problems. This is mainly because the algorithm runs best with well-defined state descriptions (such as positions, velocities) and for states where observation and time-propagation models are also well-understood. In this paper, the KF is used to fuse the positions and velocities of a humanoid robot to achieve an overall complete description of the joint positions with high accuracy and fewer uncertainties.

Sensor fusion can enable to obtain more accurate demonstration data, while effective learning methods can learn more desired features of data. A deep learning neural network, as a kind of popular feature learning algorithm, has been successfully applied in various fields because of its powerful approximation capability (Ciresan et al., 2012; Marblestone et al., 2016; Sze et al., 2017). Although this advantage makes it apply in amounts of areas, it often needs a large number of datasets to train the network. Due to this, a complicated network structure is needed to deal with them, and then the network will suffer from a time-consuming process. Apart from that, the network is also faced with the issue that entire retraining when new samples are inputted. Considering these problems of deep structure learning methods, Chen and Liu (2017) proposed an incremental learning method, which provides an alternative way for deep structure neural network (Liu and Chen, 2017). The incremental learning network can rapidly learn and model the target system without a retraining process if new samples are fed into it. Also, the structure of this network can be expanded flexibly in a wide sense. Like a deep structure neural network, the approximation capability of an incremental learning network is universal (Chen et al., 2018). Hence, it has been successfully engaged in different fields employing efficient modeling and fast learning ability. These applications mainly involved two aspects: classification and regression. Most researchers employ this algorithm in various kinds of classification (Zhang et al., 2018; Li J. et al., 2019). For example, Zhang et al. (2018) applied it to recognize facial expression to improve the accuracy of recognition. Based on this method, Wang et al. (2018) integrated it with the convolution neural network to classify EEG emotion which achieves the highest average recognition accuracy. The applications, which are involved in different curves fitting, were seldom. Luo et al. (2019) used it to estimate human intention by predicting the force of human hand. Chen et al. (2018) have proved that compared with function approximation and time series prediction, the incremental learning algorithm is superior to other learning algorithms, such as SVM, least squared

SVM, and extreme learning machine (ELM), on regression performance. It is noted that the incremental learning network and ELM algorithm (Huang et al., 2004) are similar among these methods. Both networks have the structure of a single layer. Also, both networks are thought to have potential advantages in learning rate and generalization ability (Huang et al., 2006; Chen et al., 2018). Apart from that, the incremental learning network can be employed in other scenarios, such as fault diagnosis (Zhao et al., 2019) and monkey oculomotor decision decoding (Shi et al., 2020). However, this method is seldom used in HRI to improve the performance of robot learning.

For the TbD system, we can teach a robot to move as the desired trajectory. However, human movement is not always necessarily optimal for the robot when it tries to repeat and accomplish a task. Therefore, teaching a robot remotely, there will be some deviations between the robot's trajectory and the target trajectory. Through learning based on a neural network, the robot's trajectory can approach the target trajectory. To achieve that, the incremental learning algorithm is used to learn the fused features of a certain task from different sensors to enhance the learning performance. Then, experiments are performed to verify the effectiveness of the proposed multimodal framework.

The main contribution of this paper is to develop a framework that integrates the advantages of the multiple modal information fusion with the approximation capability of the incremental learning algorithm to enhance the performance of the TbD system. The remainder of the paper is organized as follows. The System Outline section presents the whole architecture of the proposed framework. The details of the data collection, preprocessing, and learning methods are introduced in the Methodology section. The Experiments and Results section describes the experimental settings and explains the results of the experiments. The experimental results are discussed in the Discussion section. The Conclusions and Future Work section concludes this work.

SYSTEM OUTLINE
System Description
The proposed framework of the TbD is shown in **Figure 1**, which consists of three modules: the human demonstration module, the learning module, and the robot execution module.

Human demonstration module: This module, which is a virtual demonstration system, allows the human demonstrator to control the Baxter robot in Virtual Robot Experimentation Platform (V-REP) *via* human joint information. The human joint information including joint angles and joint angular velocities is recorded by the Kinect sensor and Myo armbands separately.

The learning module: This module includes two steps: data preprocessing and incremental learning. The target of data preprocessing is to align the time series information of the demonstrated tasks in the same timeline. After that, an incremental learning method is used to learn the preprocessed data.

The robot execution module: The main function of this module enables the robot to complete the task with the learned

data from the training module. To this end, a specific task will be performed by a robot to verify the effectiveness of the proposed framework.

System Principle
The principle of the overall system based on the proposed method with multimodal sensor data fusion is presented in **Figure 2**. As shown in **Figure 2**, it consists of a Kinect sensor, two Myo armbands, and a Baxter robot. Kinect sensor is a motion capture device which is used to capture the motion of the human body. Myo armband, as a wearable device, is used to capture the human joint angular velocities. Baxter is a versatile semi-humanoid robot which is equipped with several advanced sensing technologies (including force, position, and torque sensing) which allow it to be applied in scientific research. V-REP is a powerful open-source robot simulator with an integrated development environment, distributed control architecture, and rich user interface to make it be an ideal platform for robot simulations. The remote application programming interface (API) in V-REP can control the robot simulation from an external application or remote hardware. This work will simulate the Baxter robot and control it by two developed API clients in V-REP.

Figure 3 shows the communication links of the virtual TbD system. It is noted that the data collected from the Kinect sensor and Myo armbands are separately recorded by two computers. Two sensors both can recognize human hand gestures. To capture the joint angles and angular velocities simultaneously, the hand state is used to control the start or end of the data collecting. When the human demonstrator's hand state is open, the data of joint angles and angular velocities will be recorded and saved in different files. Instead, the data collecting work will stop.

According to the designed human demonstration model, joint angles and angular velocities are recorded from the multiple demonstrations based on a specific task. Then, the raw demonstration data will be preprocessed and learned by the robot learning module. After that, the learned data are transferred to the Baxter robot in V-REP by MATLAB for execution. We can verify the effectiveness of the proposed method by the execution result of the Baxter robot.

METHODOLOGY

The proposed incremental learning framework includes three processes: data collection, data preprocessing, and data learning method, which correspond to the three modules mentioned above. In this section, the data collection and preprocessing processes will be introduced, and the details of the incremental learning network and the multi-sensor fusion algorithm KF also will be given.

Data Collection
In this section, we will introduce how to capture the human joint angles and angular velocities using a Kinect sensor and two Myo armbands in detail, respectively.

FIGURE 1 | Outline of the robot teaching by demonstration (TbD) system.
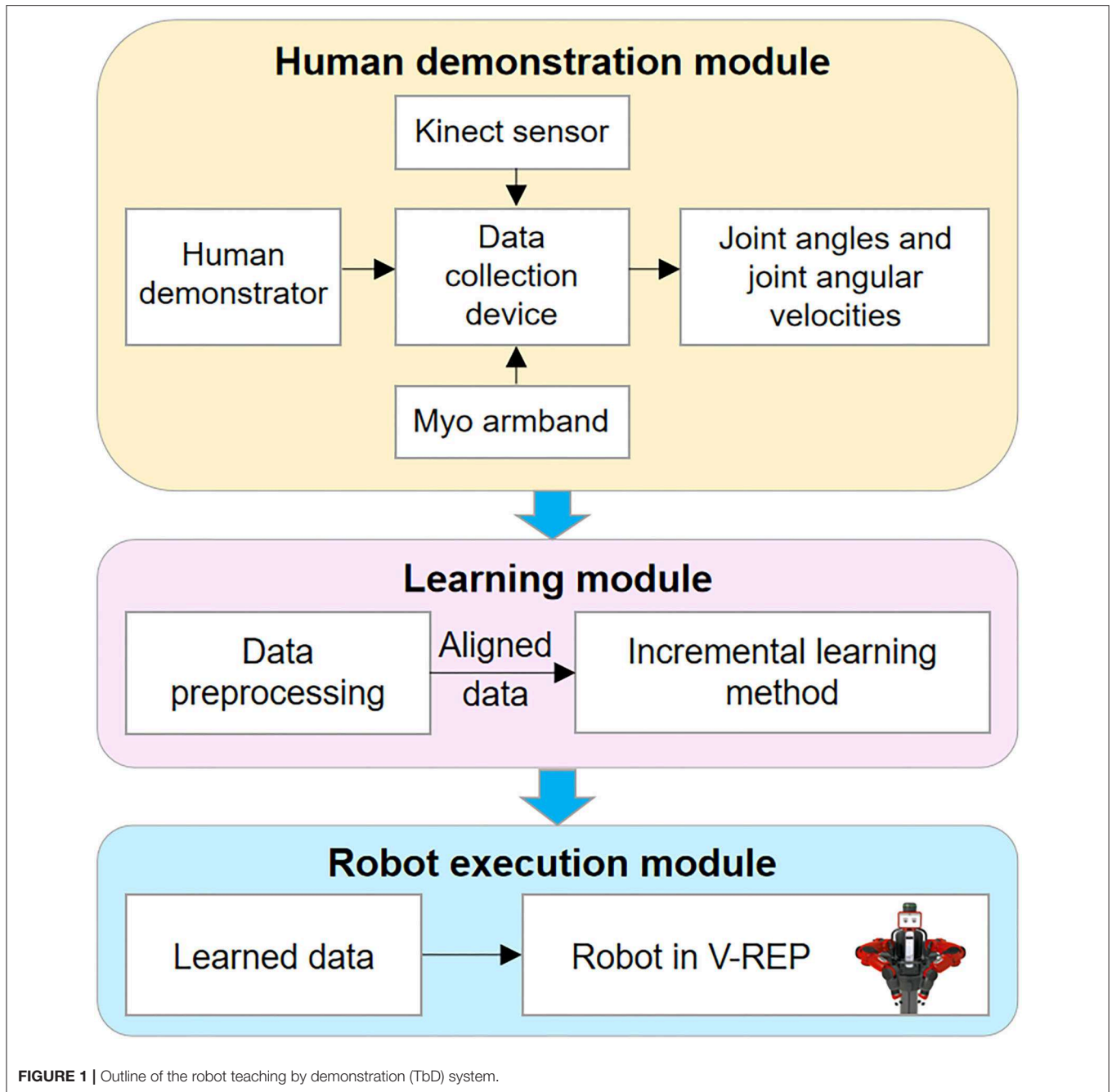
Calculation of Joint Angles Using the Space Vector Approach

Since we can get the three-dimensional (3D) joint coordinates of a human body using the Kinect sensor, the key to obtain the joint angles is how to convert these coordinates into corresponding angles. This problem can be addressed by the space vector approach. As we know, the distance between two specified 3D points $A(x_a, y_a, z_a)$ and $B(x_b, y_b, z_b)$ can be calculated by the following equation:

$$d_{AB} = \sqrt{(x_b - x_a)^2 + (y_b - y_a)^2 + (z_b - z_a)^2} \qquad (1)$$

Essentially, the distance d_{AB} is equal to the norm of the vector $\overrightarrow{AB} = (x_b - x_a, y_b - y_a, z_b - z_a)$. In a 3D space, the law of cosines can be used to calculate the angles between two known vectors. In the Kinect coordinate, a joint can be expressed as a vector. So, the angle between joint 1 ($\overrightarrow{PO}$) and joint 2 ($\overrightarrow{OQ}$) can be computed as:

$$\cos\left(\overrightarrow{PO}, \overrightarrow{OQ}\right) = \frac{\overrightarrow{PO} \cdot \overrightarrow{OQ}}{\left|\overrightarrow{PO}\right| \cdot \left|\overrightarrow{OQ}\right|} \qquad (2)$$

FIGURE 2 | The principle of the whole system. The multimodal signals are collected by the corresponding sensors from human demonstration. Two sensors are connected to different computers. Then, through incremental learning, the demonstration data are transmitted to the Baxter robot through the remote application programming interface (API) indirectly. The Baxter robot is connected to the development workstation directly. Thus, the robot can execute the demonstrated task.

FIGURE 3 | The communication links of the robot virtual demonstration system. **(A)** The communication of data collecting. **(B)** The communication of robot simulation.

According to Equation (1), we can transform the coordinates returned by Kinect into corresponding vectors. Then, the angles of these vectors can be calculated by Equation (2).

The models of the full human body and the left arm are shown in **Figure 4**. The coordinate system of Kinect in Cartesian space is constituted by three directed straight lines AX, AY, and AZ, where point A is the origin of the coordinate. According to Equation (2),

the shoulder pitch angle $\angle AOC$ can be calculated by the vectors $\overrightarrow{OA}$ and $\overrightarrow{OC}$ from the position coordinates of points A, O, and C. The elbow pitch angle $\angle OCD$ is calculated using the same method. We can get the shoulder yaw angle $\angle JAK$ in a similar way. The difference is that the vectors $\overrightarrow{AJ}$ and $\overrightarrow{AK}$ are obtained by projecting vectors $\overrightarrow{OB}$ and $\overrightarrow{OC}$ to the XZ plane.

FIGURE 4 | The models of human body skeleton and human left arm.
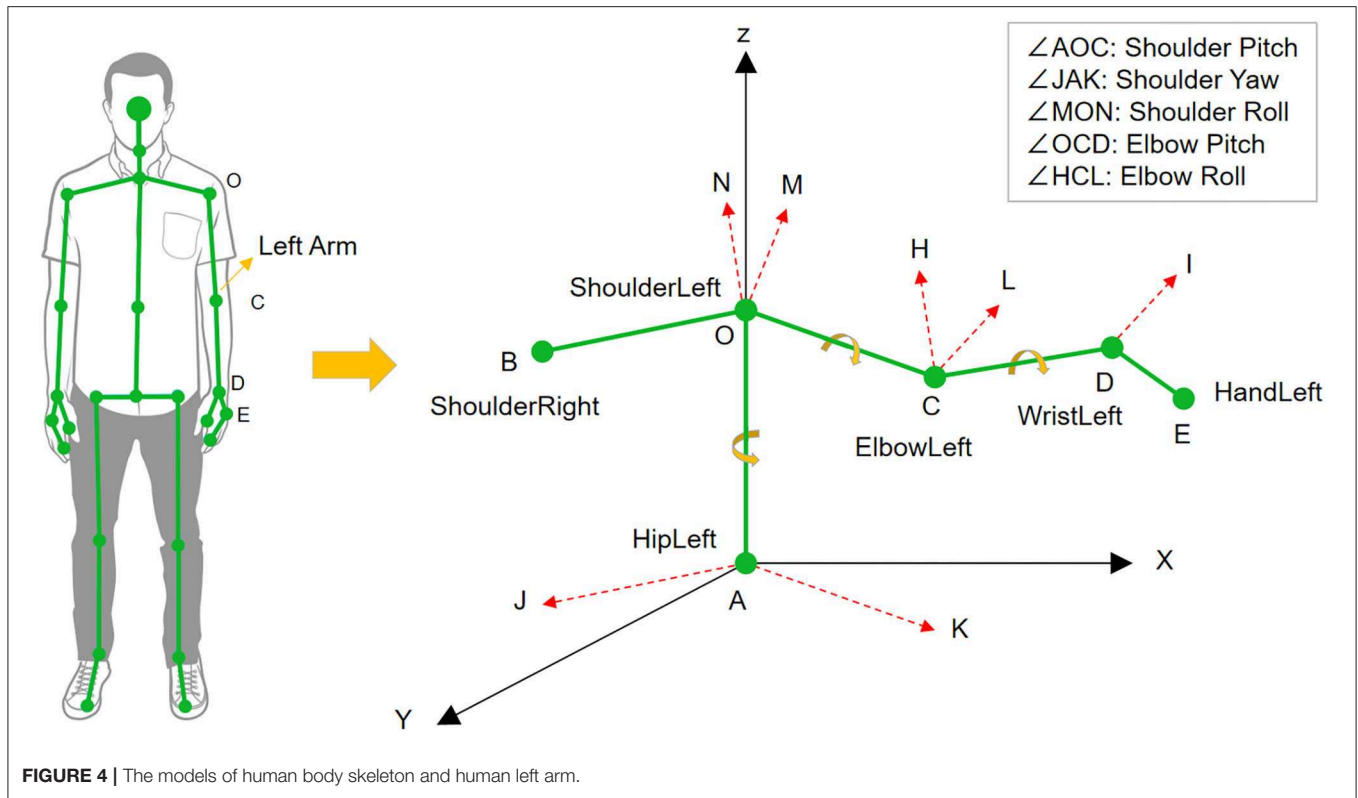
To calculate the shoulder roll angle, the cross product is applied to get the normal vector of different planes. The normal vectors of the BOC and OCD planes can be calculated by:

$$\begin{cases} \overrightarrow{OM} = \overrightarrow{OB} \times \overrightarrow{OC} \\ \overrightarrow{CH} = \overrightarrow{CO} \times \overrightarrow{CD} \end{cases} \quad (3)$$

Then, translating the vector $\overrightarrow{CH}$ along the vector $\overrightarrow{CO}$ to point O can get the vector $\overrightarrow{ON}$. So, the calculation of the shoulder roll angle $\angle MON$ is addressed. Using the same method, we can get elbow roll angle $\angle HCL$, which is the angle between the planes of OCD and CDE.

Here, only three joint angles of the human arm involving the human shoulder and elbow are collected.

Calculation of Joint Angular Velocity From Myo Armband

To obtain the joint angular velocity, two Myo armbands are needed to wear on the user's upper arm and forearm. The quaternion method is used to obtain the joint angles. Then, the joint angular velocities can be computed based the difference of the joint angles. According to Yang et al. (2018), we can assume that the joint angle of the initial position is zero. When the user's arm is moved from a pose T to a new pose P, the angle from pose T to P is the rotation angle. For the pose P, pose T can be regarded as the initial pose, and the rotation angel is the joint angle.

Assume that the Myo armband's orientation is expressed by frame (x_0, y_0, z_0) in the initial position, the current orientation is expressed by frame (x_1, y_1, z_1). Then, the angular velocities of the shoulder roll, shoulder yaw, and shoulder pitch can be obtained by the forearm armbands. The velocities of the elbow roll and pitch angles are acquired by the armbands worn on the upper arm. Thus, five joint angular velocities are obtained for each arm from a pair of Myo armbands.

Thus, we can obtain two different modalities information of human arm. After that, these joint angles and the joint angular velocities will be fused by the KF algorithm.

Data Preprocessing

The demonstration data from the Kinect sensor and Myo armband will be preprocessed before they are fed into the incremental learning method. Firstly, the data fusion method based on the KF is used to fuse the joint angles and joint angular velocities to obtain a more accurate and smooth dataset. Since the demonstration data are not matched in the timeline, then the dynamic time warping (DTW) algorithm is applied to align them. Here, the two preprocessing methods will be introduced briefly.

Data Fusion by Kalman Filter

KF, as one of the most powerful sensor fusion algorithms, can smooth noisy input data and optimize the estimation of the current state based on current measurements and the previously estimated state. These current measurements are often multiple sequential measurements from several sensors with noise. The

existing works have proved that the estimate of the system's state from multiple sensors is better than the estimate obtained from only one sensor (Gui et al., 2017). Therefore, the sensor fusion based on the KF is used to improve the accuracy of data.

This algorithm uses a series of state prediction and measurement update steps to update the state of the target object. The prediction and update steps are presented below. For a continuous simplified linear system, the dynamic model is described as follows (Davari et al., 2016):

$$
\begin{aligned}
\dot{x}(t) &= Fx(t) + Gu(t) + Mn(t) \\
z(t) &= Hx(t) + v(t)
\end{aligned}
\tag{4}
$$

where $x(t) \in \mathbb{R}^n$ is the state vector, $u(t) \in \mathbb{R}^m$ is the deterministic input vector, $z(t) \in \mathbb{R}^p$ is the measurement vector, $n(t) \in \mathbb{R}^q$ is the white noise term for the state vector with zero-mean and covariance S, and $v(t) \in \mathbb{R}^p$ is the noise term for measurement vector with zero-mean and covariance R. $F \in \mathbb{R}^{n \times n}$ and $G \in \mathbb{R}^{n \times n}$ are both system matrices. M and H are parameter matrices related to the noise and measurement, respectively. The KF model of the linear system can be expressed by the following equations (Simon, 2006):

$$
\begin{aligned}
\dot{\hat{x}}(t) &= F\hat{x}(t) + Gu(t) + K(t)\left[z(t) - H\hat{x}(t)\right] \\
K(t) &= \Sigma(t)H^T R^{-1} \\
\dot{\Sigma}(t) &= F\Sigma(t) + \Sigma(t)F^T + MSM^T - \Sigma(t)H^T R^{-1}H\Sigma(t)
\end{aligned}
\tag{5}
$$

where $K(t)$ is the filter gain, $\dot{\hat{x}}(t)$ is the state estimation of x, and $\Sigma(t)$ is the estimation of covariance.

For the above equations, we assume that $x(0)$, n, and v are uncorrelated to each other, and all the KF parameters are first order. If each joint of human arm is considered separately, we have $F = 0$, $G = 1$, $M = 1$, and $H = 1$. Thus, Equations (4, 5)

can be simplified as:

$$
\begin{aligned}
\dot{x}_i &= u_i + n_i \\
z_i &= x_i + v_i
\end{aligned}
\tag{6}
$$

$$
\begin{aligned}
\dot{\hat{x}}_i &= u_i + K(t)\left[z_i - \hat{x}_i\right] \\
K &= \Sigma R^{-1} \\
\dot{\Sigma} &= S - \Sigma R^{-1}\Sigma
\end{aligned}
\tag{7}
$$

where u_i is the ith joint angular velocity of the human arm, z_i is the ith joint position (or the joint angles), and $\hat{x}_i$ is the fused data of the ith joint. Note that the parameters K, Σ, R, and S are scalar values.

Data Preprocessing With Dynamic Time Warping

Through the human demonstration module, the angles and angular velocities of the human joints are collected from multiple demonstrations. As aforementioned, the time for every demonstration is not the same. We employ the DTW algorithm to align them in the same timeline.

DTW is a method to measure the similarity of two time series with different lengths. It has been widely used in processing the temporal sequences of video, audio, and graphics data. If two given temporal sequences g and k satisfy the boundary, monotonicity, and step size conditions, the objective of DTW can be transformed into the optimal match path problem between the two sequences. We expressed this optimal match path as:

$$
DTW(y_1, y_2) = \min(d(y_1, y_2))
\tag{8}
$$

where $d(y_1, y_2)$ represent the distance between sequences y_1 and y_2. Then, the Dynamic programming is used to solve Equation (8). At the same time, an accumulated cost matrix E with the dimension of $m \times n$ is generated. The expression of matrix E is

FIGURE 5 | The architecture of incremental learning network.
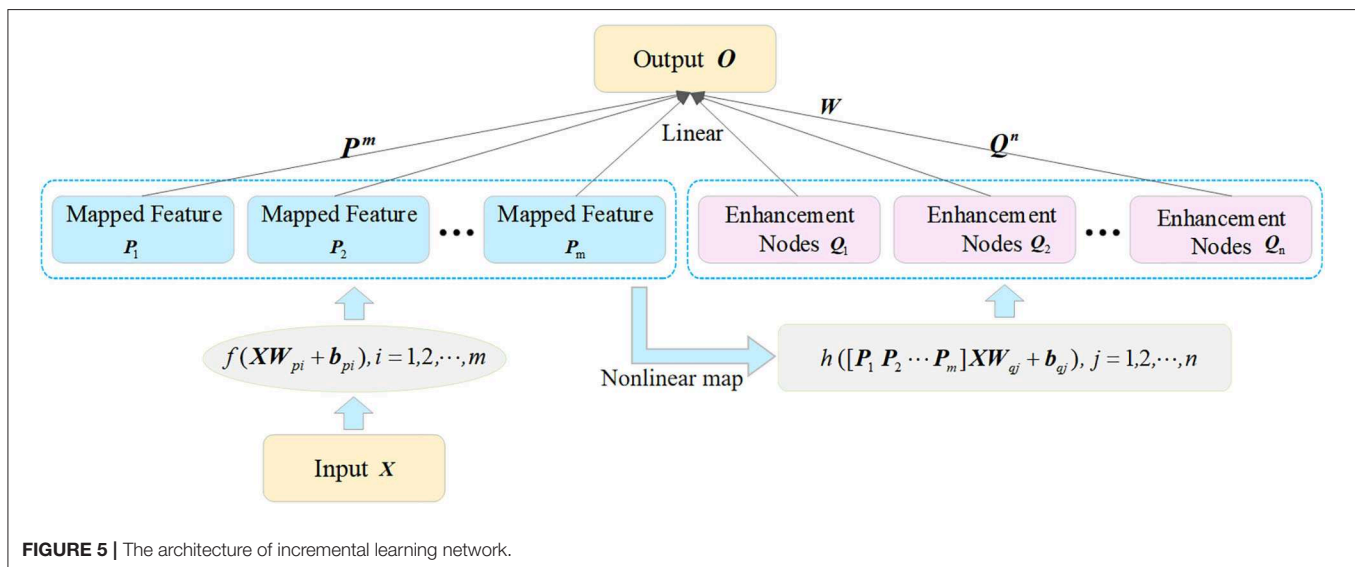
written as follows:

$$E\left(l_1, l_2\right) = \left(y_1, y_2\right) + \begin{cases} 0 & \text{if } l_1 = 1 \text{ and } l_2 = 1 \\ E\left(l_1, l_2 - 1\right) & \text{else if } l_1 = 1 \text{ and } l_2 > 1 \\ E\left(l_1 - 1, l_2\right) & \text{else if } l_1 > 1 \text{ and } l_2 = 1 \\ \min(E\left(l_1, l_2 - 1\right), \\ \quad E\left(l_1 - 1, l_2\right), \\ \quad E\left(l_1 - 1, l_2 - 1\right)) & \text{otherwise} \end{cases} \quad (9)$$

where l_1 and l_2 are the length of the sequences y_1 and y_2, respectively.

Incremental Learning Method

The incremental learning algorithm is essentially a single-layer neural network that the structure can be dynamically expanded in a wide sense. It is constructed based on the random vector functional link neural network (FLNN). The architecture of the incremental learning network is shown in **Figure 5**.

The input of this network is composed of two parts: the mapped features and the enhancement nodes. As shown in **Figure 5**, the original inputs are first transformed into a group of mapped features to extract the random features by some linear feature mappings. Then, the mapped features are extended to enhancement nodes by non-linear mappings. Further, the mapping features and the enhancement nodes in the input layer are both connected with the output linearly. Thus, the weights between the input layer and the output layer can be calculated by the ridge regression of the pseudo-inverse method.

The detailed process of the incremental network is presented as follows. For a given input dataset $\{X\}$ and m feature mapping function f_i, $i = 1, 2, \cdots, m$, the ith mapped features can be calculated as:

$$P_i = f_i\left(XW_{p_i} + b_{p_i}\right), i = 1, 2, \cdots, m \quad (10)$$

where $X \in \mathbb{R}^{m \times n}$, m is the number of training samples; n is the size of each training sample; both the bias unit b_{p_i} and the weights W_{p_i}, which connect the original input and the mapped features, are randomly generated. It is noted that the functions f_i and f_l are equal for $i \neq l$. We denote the first ith groups of mapped features as $P^i \equiv [P_1\ P_2\ \cdots\ P_i]$ and express the non-linear mappings connected the mapped features with enhancement nodes as h_j, $j = 1, 2, \cdots, n$. Then, using the non-linear function h_j, the relationship between the mapped features P_i and Q_j, the enhancement nodes can be built. The jth group of enhancement nodes is expressed as:

$$Q_j = h_j(P^m W_{p_j} + b_{p_j}), j = 1, 2, \cdots, n \quad (11)$$

where W_{p_j} and b_{p_j} are randomly generated, and W_{p_j} are the weights connecting the mapped features and the enhancement nodes. Likewise, the first jth group of enhancement nodes is denoted as $Q^j \equiv [Q_1\ Q_2\ \cdots\ Q_j]$. The enhancement nodes P_i together with the mapped features Q_j form the actual input of the incremental learning network $A \equiv [P_1, \cdots, P_m, Q_1, \cdots, Q_n] = [P^m\ Q^n]$. Hence, the output O of this network is computed as:

$$O = AW_m^n, \quad (12)$$

where the weights W_m^n connect the output layer and the input layer. Since the target output O is given, we can calculate the weights W_m^n as follows:

$$W_m^n = A^+ O, \quad (13)$$

Here, the rigid regression learning algorithm is used to solve the pseudo-inverse A in Equation (13). According to this algorithm, the pseudo-inverse A is obtained by the following equation:

$$A = \lim_{\lambda \to 0} (\lambda I + AA^T)^{-1} A^T, \quad (14)$$

Algorithm 1 presents the whole training process of the incremental learning network.

Algorithm 1 The procedure of the incremental learning network.

Input: Demonstration dataset X, mapped feature group m, and enhancement nodes group n.
Output: The parameter matrix W.
for $i = 1$ to m **do**
    Randomly initialize the weights W_{p_i} and bias unit b_{p_i};
    Calculate P_i according to Equation (10).
end
Set mapped features group $P^m \equiv [P_1\ P_2\ \cdots\ P_m]$;
for $j = 1$ to n **do**
    Randomly initialize the weights W_{p_j} and bias unit b_{p_j};
    Calculate Q_j according to Equation (11).
end
Set the enhancement nodes group $Q^n \equiv [Q_1\ Q_2\ \cdots\ Q_n]$;
Calculate weights W_m^n according to Equation (13).

As aforementioned, the ELM method and the incremental learning method both are single-layer neural networks, and the learning speed of two methods is also fast. For the incremental learning network, if the learning cannot reach the desired result, it can be addressed by inserting additional enhancement nodes in a wide sense not deep way to achieve a better performance. The increase of the enhancement nodes will result in the recalculation of weights. It is worth noting that only a part of the weights needs to be recalculated, not all weights. The new weights are calculated by the following equations:

$$W_m^{n+1} = \begin{bmatrix} W_m^n - DB^T O \\ O \end{bmatrix}, \quad (15)$$

where $C = h_{n+1}\left(P^m W_{p_{n+1}} + b_{p_{n+1}}\right) - A^n D$, $D = (A^n)^+ h_{n+1}(P^m W_{p_{n+1}} + b_{p_{n+1}})$, and

$$B^T = \begin{cases} (C)^+ & \text{if } C \neq 0 \\ (1 + D^T D) B^T (A^n)^+ & \text{if } C = 0 \end{cases}, \quad (16)$$

Note that 0 is zero matrix, and O is the output of the network.

For the ELM network, the solution to improve performance is to increase the number of hidden layer neurons, which

results in more connecting parameters. Thus, a great number of parameters including all weights need to be updated. It means that the ELM network suffers from a complete relearning process. In this respect, the incremental learning network is different. Besides, the incremental learning network is allowed to increase the number of input samples without relearning all samples. Likewise, only the newly added samples need to be learned by the incremental learning network. It also implies that the incremental learning network can adapt to new data without forgetting its existing knowledge, instead of relearning all samples. This is the difference in the structural expansion between the two networks.

Furthermore, the mapped features of the incremental learning network are randomly generated from the original input dataset {X}. In other words, the mapped features are the results of feature representation for the original input data. Feature representation can capture the efficient characteristics of the data to achieve outstanding performance in supervised learning tasks (Chen et al., 2018). It explains why the incremental learning network can learn the desired features. Also, it shows that the actual input data of the two networks are different. This implies the difference between the two networks from another aspect.

As stated above, the motivation to use the incremental learning algorithm is its convenience in a specific scene and feature learning ability.

EXPERIMENTS AND RESULTS

Experimental Setup

We test our method on the Baxter robot. The experimental system is shown in **Figure 6**. The hardware devices consist of a Baxter robot, a Kinect sensor, and two Myo armbands. Based on

the platform, two tasks (wiping and pushing) are performed to verify the effectiveness of the proposed TbD system.

In the wiping task, the robot in V-REP follows human motion to raise his left arm, move toward the left, and then put it down along the path it passed. The difficulty of this task is that the trajectories of up and down motions should be consistent. The repetitive processes with the same task are performed more than 16 times.

The wiping task is performed under the following three conditions:

- *Condition 1: with Kinect sensor data and incremental learning method.* The demonstration data are only collected from the Kinect sensor but without Myo armbands. Through processing of DTW, the incremental learning network is used to learn it. There is no data fusion in this condition.
- *Condition 2: with two sensors data (Kinect and Myo armband) and incremental learning method.* The demonstration data are collected from both Kinect and Myo armbands. In this case, the sensor fusion process is added before data preprocessing with DTW algorithm. Later, the preprocessed data are learned by the incremental learning network.
- *Condition 3: with two sensors data (Kinect and Myo) and ELM algorithm.* The demonstration data collection and processing processes are the same as the second condition. The difference is that the learning method of these data is ELM algorithm (Huang et al., 2004) instead of the incremental learning network.

In summary, the first condition is to show the performance of incremental learning network with only joint angle information but without joint angular velocities. The second condition is set
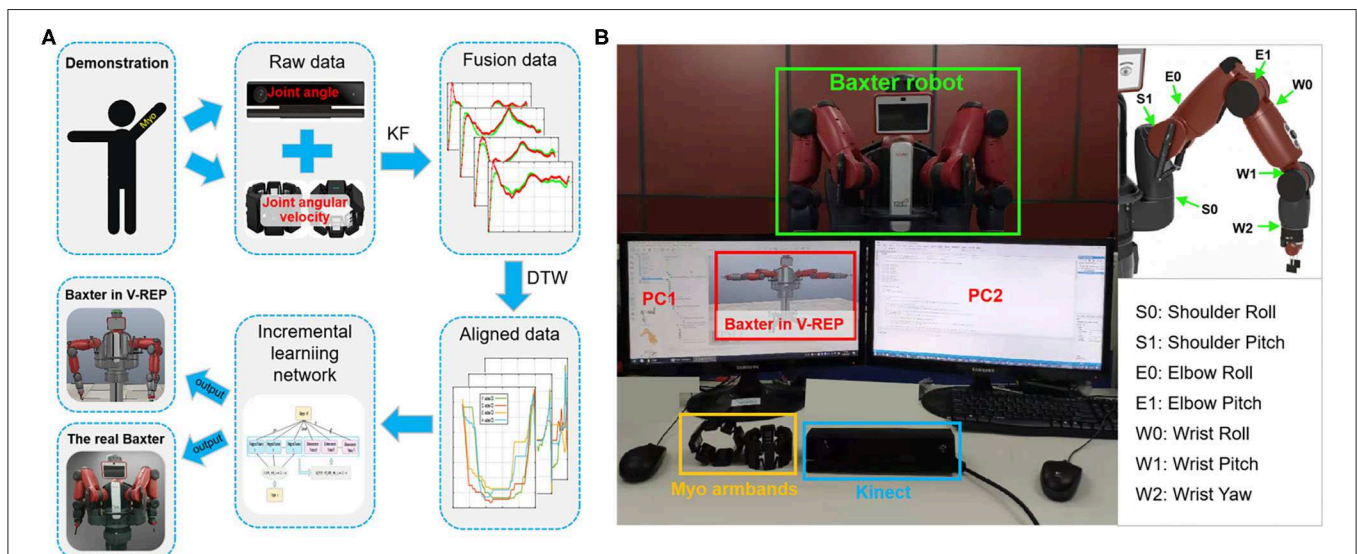
FIGURE 6 | The experimental system. **(A)** The diagram of the experimental system. **(B)** The experimental platform of the demonstration phase. During demonstration, the joint angles and joint angular velocities of the human arm are collected simultaneously by Kinect and Myo armband. Then, the raw demonstration data will be fused and aligned in the same timeline by the Kalman filter (KF) and dynamic time warping (DTW) algorithms in turn. After that, the incremental learning network is applied to learn the processed data. During the robot learning phase, the learned data are directly sent to the robot model in Virtual Robot Experimentation Platform (V-REP) and the real Baxter robot.
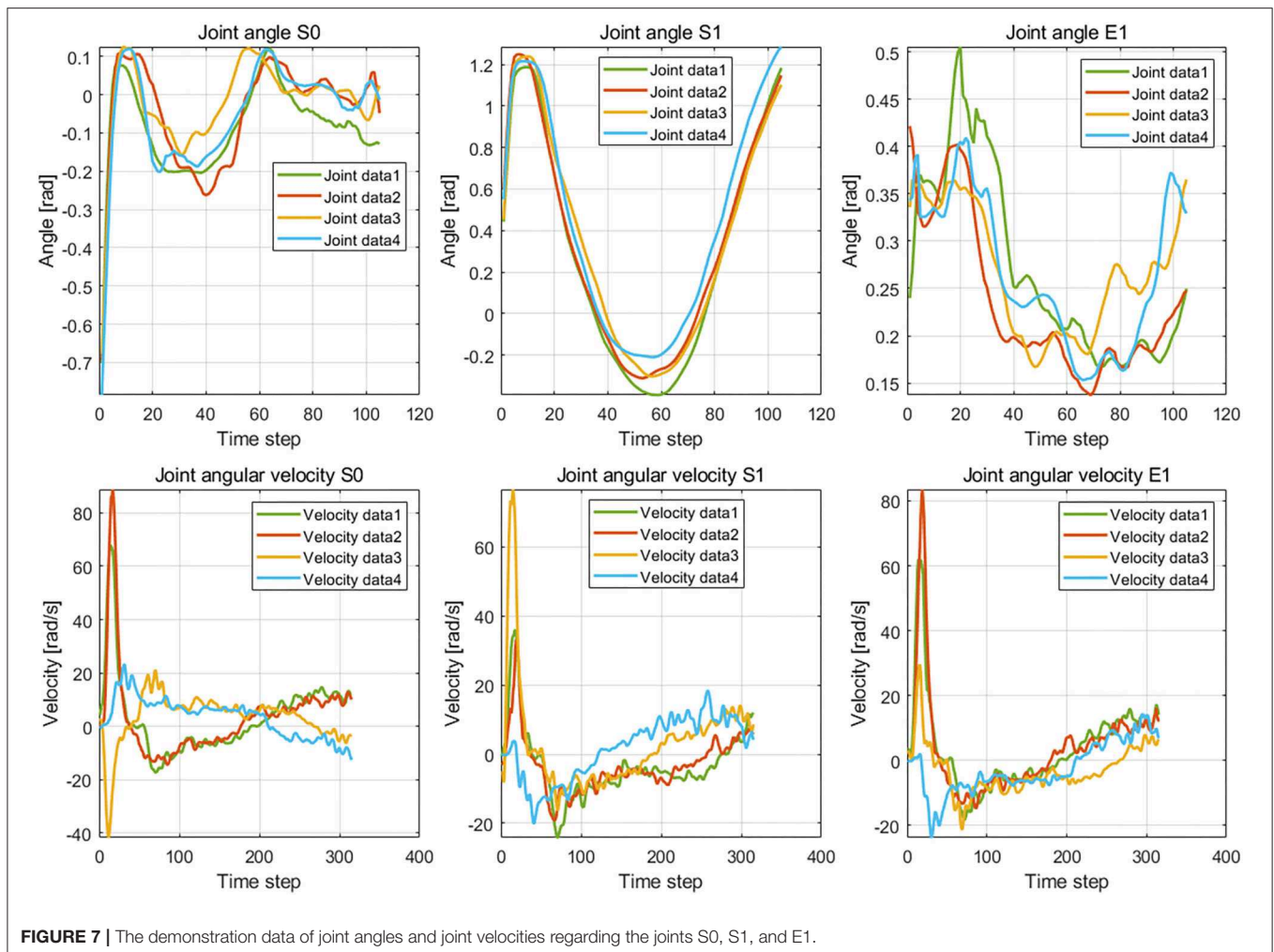
FIGURE 7 | The demonstration data of joint angles and joint velocities regarding the joints S0, S1, and E1.

to validate the proposed incremental learning framework with sensor fusion, while the third condition is to test the performance of the ELM network with sensor fusion.

To find the optimal number of feature mapping group m and enhancement nodes group n, we change m and n from 1 to 50 for the incremental learning network. The result shows that the highest accuracy appears when m and n are 6 and 8, respectively. A similar test is conducted for the ELM algorithm. We can get that when the number of hidden layer neurons is 11, the ELM network has the best accuracy.

Experimental Results
Results of the Wiping Task

The results of the wiping tasks are shown in **Figures 7–10**. In the demonstration phase, the raw multimodal data are recorded by different sensors. **Figure 7** presents four randomly selected samples of human demonstrations. The results of preprocessing are shown in **Figures 8**, **9**. **Figure 8** shows the curves of the fusion datasets. Note that there are deviations between the raw joint angles and the fused data. **Figure 9** displays the aligned results of the fused datasets. Compared with the demonstration data without alignment in the timescale, the aligned data also retain

the primary characteristics through the aligning process by DTW algorithm. The aligned results prepare for the next training and learning of the neural network.

The difference between the first and second conditions is that the demonstration samples are different for the incremental learning network. Since deviations between the raw original joint angles and the fusion joint angles exist, the results of DTW aligning will be different. We can observe it from the images of the second and third rows in **Figure 9**. The dimension of the original raw demonstration data is 105. After processing by DTW, the dimensions of these datasets are 367 and 355 for the raw data and fusion data, respectively.

The trajectories learned by the incremental learning network and ELM network are shown in **Figures 10A–C** with red dotted lines. For the Baxter robot, all changes of seven joint angles are aimed to obtain a desired trajectory of the end effector because the robot execution eventually depends on the end effector. We recorded the trajectory of the end effector in Cartesian space during robot execution, which is shown in **Figure 10D**. Seven joint angles of the real Baxter's left arm are also recorded and shown in **Figure 10E**.
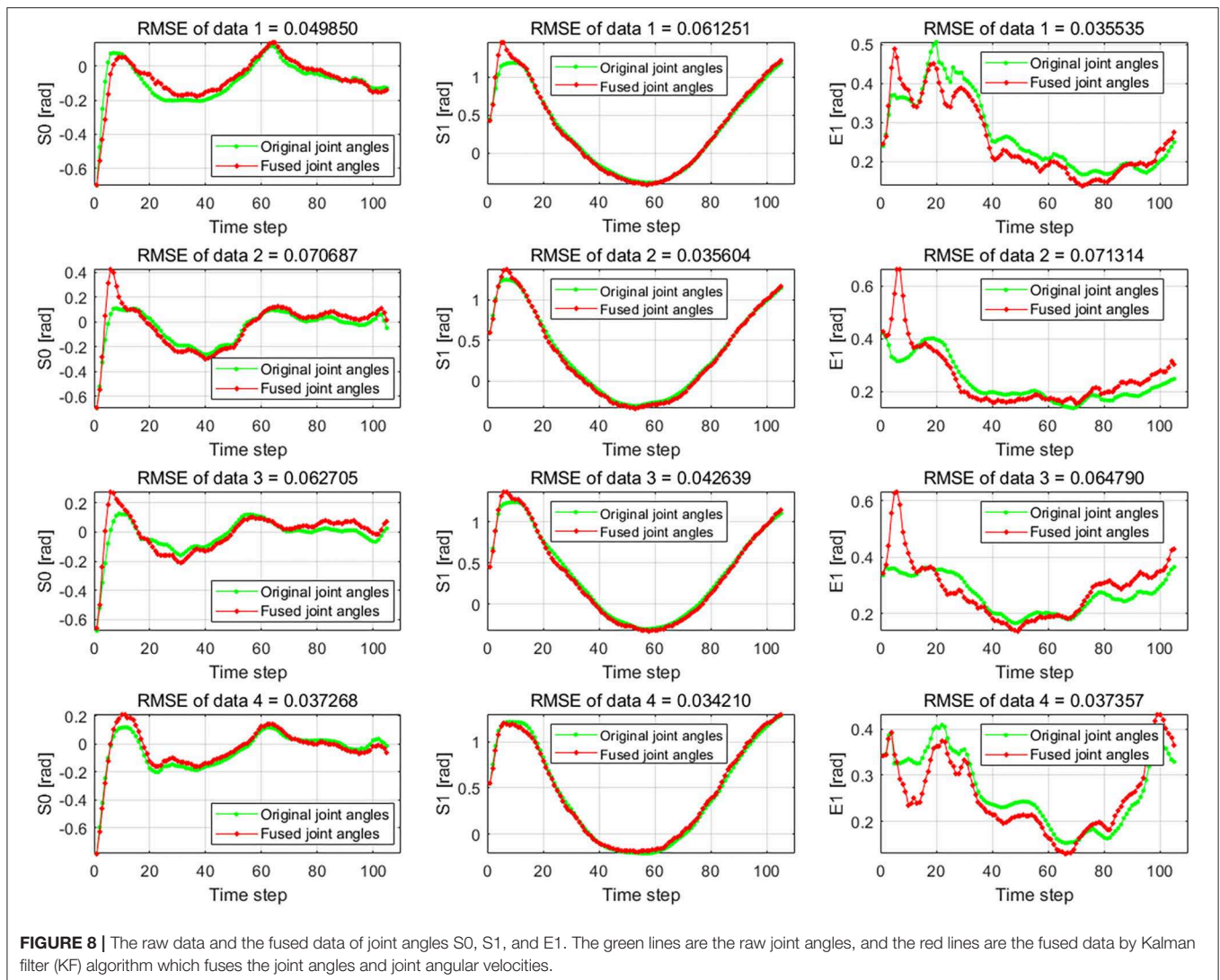
FIGURE 8 | The raw data and the fused data of joint angles S0, S1, and E1. The green lines are the raw joint angles, and the red lines are the fused data by Kalman filter (KF) algorithm which fuses the joint angles and joint angular velocities.

Based on the learned results by the incremental learning network, the Baxter robot can implement the wiping task. The robot implementation includes a simulation experiment of the Baxter robot in V-REP and an experiment for real Baxter robot. And the wiping task covers four directions of continuous and smooth movement: up, down, right, and left.

Results of the Pushing Task

To test the generalization ability of the proposed method, a pushing task is performed. The pushing task requires the robot to push two square workpieces over on the desk in sequence. In other words, the robot should firstly push the workpiece on the right to the desk. During pushing, the robot cannot touch the workpiece on the left. Then, the robot pushes the right one to the desk. The short distance between the two workpieces makes it more difficult for the robot to complete this task. Because the aims of the pushing task and the wiping task are different, the pushing task is only conducted under the above conditions

2 and 3. The experimental steps are the same as the wiping task. The experimental results are ultimately reflected in the trajectory of the robot end effector, which determines whether the robot can complete the demonstrated task. Hence, **Figure 11** only presents the trajectories of the real Baxter robot in the simulation scene and real environment, but not the results of data preprocessing.

As shown in **Figure 11**, the distance between the two tasks is very close. Any deviation in the trajectory of the robot end effector could result in the robot failing to complete the task. Nevertheless, we can find that the Baxter robot can complete the pushing task well from the experimental results. For this task, the results can directly reflect the performance of the two learning methods. Since the result of the robot end effector's trajectory under condition 3 is a failure to complete the pushing task, the corresponding result is not displayed in **Figure 11**. Also, the results for the wiping tasks are undiscussed in the next section. These results illustrate that the proposed method can not only improve the performance of TbD but also be applied in learning

FIGURE 9 | The fused data and the aligned data by the dynamic time warping (DTW) algorithm. The figures in the first line and second line show the result of the raw demonstration data fused by the Kalman filter (KF) method and alignment by the DTW algorithm, respectively. The images in the third line display the aligned result of the data collected by the Kinect sensor. The three columns are the corresponding results of joints S0, S1, and E1, respectively.

different tasks for the robot. It implies that the proposed method has good generalization ability.

DISCUSSION

The purpose of this work is to investigate the practical effect of the proposed method on robot TbD, as well as to explore the impact on the result considering the fusion of multiple modality information. It is noted that only the results of the wiping task are discussed in this section. Because the pushing task requires more accurate execution for the Baxter robot, the performance of the learning method can be directly judged by the execution results of the Baxter robot. The experimental results of the pushing task clearly illustrate that using the incremental learning method can enable the robot to complete the pushing task well, while the ELM algorithm cannot.

Firstly, we examine the effectiveness of multimodal data fusion by comparing the results in **Figures 10A,B** under the first and

second conditions. It is clear that the bias between the reference trajectories and the real trajectories of the first condition is much larger than the second one, especially in the start phase of the interval (0, 100). And the curves of the real trajectories are inconsistent for the first and second conditions in **Figures 10A,B**. For joints S0 and S1, the trend of reference trajectories is almost the same under conditions 2 and 3. Concerning the joint E1, the differences between the curves are especially evident under the same conditions. In **Figure 10A**, the maximum difference value between the reference and the real value is already close to 2. But this value is not more than 0.6 under the second condition, which can be observed from **Figure 10B**. The trajectories of the real Baxter robot regarding the joints S0, S1, and E1 illustrate that the multimodal data fusion can promote a result that is much closer to the reference values.

Next, we discuss the superiority of the proposed method in comparison with another robot learning method. **Figures 10B,C** show the corresponding results by using the proposed method and the ELM method. The red dotted lines are the learning

FIGURE 10 | The execution results of the real Baxter robot. (A–C) The joint angles of the real Baxter robot under conditions 1, 2, and 3, respectively. (D) The trajectories of Baxter end effector under the three conditions. (E) Seven joint angles of the real Baxter robot under the three conditions. In panels (A–C), the red dotted line is the output of the network under the three conditions, and the green solid line is the real Baxter robot's joint angles. The green solid dots are the start point of the Baxter end effector, and the red solid dots are the end point in panel (D). The red, green, and purple solid lines, respectively, display the joint angles of S0, S1, and E1 in panel (E), and the other colored solid lines display the rest of the four joint angles.

FIGURE 11 | The execution results of the pushing task. The green lines in the four figures are the trajectories of the real Baxter robot. The trajectories of the real Baxter robot are sent to the Baxter model in the Virtual Robot Experimentation Platform (V-REP), which is plotted out with green lines.

FIGURE 12 | The errors of the root mean square error (RMSE) and mean absolute error (MAE) between the learned joint angles and the real Baxter robot joint values under the three experimental conditions. **(A)** RMSE. **(B)** MAE. The area of each color square indicates the magnitude of two errors.

result of the two networks. Obviously, the trends of red reference trajectories in **Figures 10B,C** are consistent. It indicates that both of the methods can learn the features of the joint angles to complete the wiping task. However, the biases between the reference trajectories and the real trajectories of the two methods are different. To analyze the result quantitatively, we calculate the mean absolute error (MAE) and root mean square error (RMSE), which are shown in the second and third columns of **Figures 12A,B**. The RMSE is calculated as follows:

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{t=1}^{N} (y_t - \hat{y}_t)^2}, \tag{17}$$

where N is the size of the demonstration sequences, $\hat{y}_t$ is the value fused by the KF algorithm, and y_t is the value measured by the Kinect sensor.

The calculation of MAE is as follows:

$$\text{MAE} = \frac{1}{N} \sum_{t=1}^{N} |y_t - \hat{y}_t|, \tag{18}$$

Noteworthy, RMSE and MAE are the errors between the reference data (namely, the output of the incremental learning network or the learned trajectories) and that of the real Baxter robot data. The areas of the squares using the proposed method are less than that of squares using the ELM method. The RMSE and MAE results imply that the errors of the incremental learning method are smaller. This shows that the experiment performance of the proposed method is better than the ELM method. We can

also find that the maximum error under the three conditions is from the result without data fusion. It also implies that through data fusion, both errors are diminished notably.

As aforementioned, the difficulty of the wiping task is how to ensure that the trajectories of upward and down motion are consistent. We find that the result under condition 1 is worst, and the trajectory is disordered in **Figure 10D**. On the contrary, the trajectories under the second and third conditions are smooth and orderly. Furthermore, the result of the second condition is better than that of the third one. It also proves that data fusion can improve the experiment performance in another way. Besides, we compute the distance between the start point and the end point for three conditions, which are 0.8825, 0.0135, and 0.0778, respectively. For the results, the shorter the distance, the better the performance. It is obvious that the distance for the second condition is the shortest. That is to say, both the trajectories of the end effector and the distance illustrate that the proposed method is better. These results suggest that the proposed method is superior to the ELM method, not only the joint angles but also the trajectories of the end effector.

Lastly, we compare all recorded joint angles of the robot's left arm under the three conditions. The desired result is that the other four joint angles are approximate to zero except for S0, S1, and E1, which is shown in **Figure 10E**. The four joint angles in the interval (0, 100) under the first condition are much bigger than zero, and then they gradually trend to zero. However, the four joint angles are much closer to zero under the second and third conditions from beginning to end. It also shows that sensor fusion can decrease demonstration errors.

To sum up, the demonstration data with multimodal information can significantly improve the experiment performance, and the proposed method can achieve a better execution result with smaller errors. This is probably because data fusion is beneficial to obtain a demonstration dataset close to the real value. At the same time, the KF algorithm smooths the raw data to some extent. All of these help the real robot move smoothly and efficiently. On the other hand, the incremental learning network can learn more effective features to enhance TbD performance.

CONCLUSIONS AND FUTURE WORK

In this paper, we propose an incremental learning framework to learn demonstration features by integrating different modality data. Using the proposed method and the KF algorithm, the TbD performance is remarkably improved. To verify the proposed method, comparative experiments involving the incremental learning network and ELM algorithm were conducted based on a Baxter robot in a real physical environment. Through the experiments, the robot achieved a better result with smaller errors using the proposed network on the basis of two modality information fusions. The effectiveness of the proposed method was verified by analyzing the learned data and the real robot data in comparison with ELM methods. As a result, the proposed method can learn more critical features to get the desired result. Since the TbD system is based on two modality information fusions, we also verify the effect of multimodal integration on the real robot. Compared with the results of single-modality data, the multimodal data with sensor fusion can achieve a better performance. It implies that the fusion of modality information is beneficial to improve the accuracy of data. To test the generalization of the proposed method, a pushing task is performed. The successful experiment results show that the proposed method has the generalization ability in TbD. In the future, integrating modality information from different types of sensors, e.g., force, will be addressed to perform complex tasks online. We will further explore the complete time of a specific task for the real robot by employing other methods. Also, how to reduce the effect of demonstrations from different people on the experimental results is taken into account.

DATA AVAILABILITY STATEMENT

The datasets generated for this study are available on request to the corresponding author upon reasonable request.

AUTHOR CONTRIBUTIONS

ACKNOWLEDGMENTS

REFERENCES

Argyrou, A., Giannoulis, C., Sardelis, A., Karagiannis, P., Michalos, G., and Makris, S. (2018). A data fusion system for controlling the execution status in human-robot collaborative cells. *Procedia CIRP* 76, 193–198. doi: 10.1016/j.procir.2018.01.012

Bijarbooneh, F. H., Du, W., Ngai, E. C.-H., Fu, X., and Liu, J. (2016). Cloud-assisted data fusion and sensor selection for internet of things. *IEEE Internet Things J.* 3, 257–268. doi: 10.1109/JIOT.2015.2502182

Billard, A., Calinon, S., Dillmann, R., and Schaal, S. (2008). "Robot programming by demonstration," in *Springer Handbook of Robotics*, eds B. Siciliano and O. Khatib (Berlin: Springer Press), 1371–1394. doi: 10.1007/978-3-540-30301-5_60

Caterina, M., Bülthoff, H. H., and Stegagno, P. (2015). "Autonomous vegetation identification for outdoor aerial navigation," in *IEEE/RSJ International Conference on Intelligent Robots and Systems* (Hamburg), 3105–3110.

Chavez-Garcia, R. O., and Aycard, O. (2016). Multiple sensor fusion and classification for movingobject detection and tracking. *IEEE Trans. Intell. Transp. Syst.* 17, 525–534. doi: 10.1109/TITS.2015.2479925

Chen, C. P., and Liu, Z. (2017). Broad learning system: an effective and efficient incremental learning system without the need for deep architecture. *IEEE Trans. Neural Networks Learn. Syst.* 29, 10–24. doi: 10.1109/TNNLS.2017.2716904

Chen, C. P., Liu, Z., and Feng, S. (2018). Universal approximation capability of broad learning system and its structural variations. *IEEE Trans. Neural Networks Learn. Syst.* 30, 1191–1204. doi: 10.1109/TNNLS.2018.2866622

Chung, H., Oh, S., Shim, D. H., and Sastry, S., S. (2011). Toward robotic sensor webs: algorithms, systems, and experiments. *Proc. IEEE* 99, 1562–1586. doi: 10.1109/JPROC.2011.2158598

Ciresan, D., Meier, U., and Schmidhuber, J. (2012). "Multi-column deep neural networks for image classification," in *IEEE Conference on Computer Vision and Pattern Recognition* (Providence, RI), 3642–3649. doi: 10.1109/CVPR.2012.6248110

Cortes, C., and Vapnik, V. (1995). Support-vector networks. *Mach. Learn.* 20, 273–297. doi: 10.1007/BF00994018

Crisan, D., and Doucet, A. (2002). A survey of convergence results on particle filtering methods for practitioners. *IEEE Trans. Signal Process.* 50, 736–746. doi: 10.1109/78.984773

Davari, N., Gholami, A., and Shabani, M. (2016). Multirate adaptive kalman filter for marine integrated navigation system. *J. Navig.* 70, 1–20. doi: 10.1017/S0373463316000801

Deng, Y., Li, F., and Xin, H. (2018). Research on multimodal human-robot interaction based on speech and gesture. *Comput. Electr. Eng.* 72, 443–454. doi: 10.1016/j.compeleceng.2018.09.014

Din, S., Ghayvat, H., Paul, A., Ahmad, A., Rathore, M. M., and Shafi,. I. (2015). "An architecture to analyze big data in the internet of things," in *9th International Conference on Sensing Technology* (Auckland), 677–682. doi: 10.1109/ICSensT.2015.7438483

Eitel, A., Springenberg, J. T., Spinello, L., Riedmiller, M., and Burgard, W. (2015). "Multimodal deep learning for robust RGB-d object recognition," in *IEEE/RSJ International Conference on Intelligent Robots and Systems* (Hamburg), 681–687. doi: 10.1109/IROS.2015.7353446

Elmenreich, W. (2002). *Sensor fusion in time-triggered systems* (Ph.D. Thesis). Vienna University of Technology, Vienna, Austria.

Ewerton, M., Arenz, O., Maeda, G., Koert, D., Kolev, Z., Takahashi, M., et al. (2019). Learning trajectory distributions for assisted teleoperation and path planning. *Front. Robot. AI* 6:89. doi: 10.3389/frobt.2019.00089

Fang, B., Wei, X., Sun, F., Huang, H., Yu, Y., and Liu, H. (2019). Skill learning for human-robot interaction using wearable device. *Tsinghua Sci. Technol.* 24, 654–662. doi: 10.26599/TST.2018.9010096

Fung, M. L., Chen, M. Z. Q., and Chen, Y. H. (2017). "Sensor fusion: a review of methods and applications," in *The 29th Chinese Control and Decision Conference* (Chongqing), 3853–3860. doi: 10.1109/CCDC.2017.7979175

Gibson, R. E., Hall, D. L., and Stover, J. A. (1994). "An autonomous fuzzy logic architecture for multisensor data fusion," in *International Conference on Multisensor Fusion and Integration for Intelligent Systems* (Las Vegas, NV), 143–150. doi: 10.1109/MFI.1994.398450

Gui, K., Liu, H., and Zhang, D. (2017). Toward multimodal human-robot interaction to enhance active participation of users in gait rehabilitation. *IEEE Trans. Neural Syst. Rehabil. Eng.* 25, 2054–2066. doi: 10.1109/TNSRE.2017.2703586

Haghighat, M. B. A., Aghagolzadeh, A., and Seyedarabi, H. (2011). Multi-focus image fusion for visual sensor networks in DCT domain. *Comput. Electr. Eng.* 37, 789–797. doi: 10.1016/j.compeleceng.2011.04.016

Hu, G. (2010). "Neural network applications in sensor fusion for a mobile robot motion," in *WASE International Conference on Information Engineering* (Beidaihe), 46–49.

Huang, G. B., Zhu, Q. Y., and Siew, C. K. (2004). "Extreme learning machine: a new learning scheme of feedforward neural networks," in *IEEE International Joint Conference on Neural Networks* (Budapest), 985–990.

Huang, G. B., Zhu, Q. Y., and Siew, C. K. (2006). Extreme learning machine: theory and applications. *Neurocomputing* 70, 489–501. doi: 10.1016/j.neucom.2005.12.126

Iyengar, S. S., Sastry, S., and Balakrishnan, N. (2003). Foundations of data fusion for automation. *IEEE Instrum. Meas. Mag.* 6, 35–41. doi: 10.1109/MIM.2003.1251481

Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. *Trans. ASME J. Basic Eng.* 82, 35–45. doi: 10.1115/1.3662552

Khaleghi, B., Khamis, A., Karray, F. O., and Razavi, S. N. (2013). Multisensor data fusion: a review of the state-of-the-art. *Inf. Fusion* 14, 28–44. doi: 10.1016/j.inffus.2011.08.001

Koshmak, G., Loutfi, A., and Linden, M. (2016). Challenges and issues in multisensor fusion approach for fall detection: review paper. *J. Sens.* 2016, 1–12. doi: 10.1155/2016/6931789

Li, C., Fahmy, A., and Sienz, J. (2019). An augmented reality based human-robot interaction interface using kalman filter sensor fusion. *Sens.* 19:4586. doi: 10.3390/s19204586

Li, J., Zhong, J., Chen, F., and Yang, C. (2019). "An incremental learning framework for skeletal-based hand gesture recognition with leap motion," in *9th IEEE International Conference on CYBER Technology in Automation, Control and Intelligent System* (Suzhou), 13–18. doi: 10.1109/CYBER46603.2019.9066761

Liu, Z., and Chen, C. P. (2017). "Broad learning system: structural extensions on single-layer and multi-layer neural networks," in *International Conference on Security, Pattern Analysis, and Cybernetics* (Shenzhen), 136–141. doi: 10.1109/SPAC.2017.8304264

Luo, J., Liu, C., Li, Y., and Yang, C. (2019). "A framework of human impedance recognition," in *25th International Conference on Automation and Computing* (Lancaster), 1–6. doi: 10.23919/IConAC.2019.8895250

Luo, R. C., and Chang, C. C. (2012). Multisensor fusion and integration: a review on approaches and its applications in mechatronics. *IEEE Trans. Ind. Inf.* 8, 49–60. doi: 10.1109/TII.2011.2173942

Marblestone, A. H., Wayne, G., and Kording, K. P. (2016). Toward an integration of deep learning and neuroscience. *Front. Comput. Neurosci.* 10:94. doi: 10.3389/fncom.2016.00094

Medjahed, H., Istrate, D., Boudy, J., Baldinger, J. L., and Dorizzi, B. (2011). "A pervasive multi-sensor data fusion for smart home healthcare monitoring," in *IEEE International Conference on Fuzzy Systems* (Taipei), 1466–1473. doi: 10.1109/FUZZY.2011.6007636

Mitchell, H. B. (2007). *Multi-Sensor Data Fusion: An Introduction.* Heldelberg: Springer Science and Business Media Press.

Shi, Z., Chen, X., Zhao, C., He, H., Stuphorn, V., and Wu, D. (2020). Multi-view broad learning system for primate oculomotor decision decoding. *IEEE Trans. Neural Sys. Rehabil. Eng.* 1. doi: 10.1109/tnsre.2020.3003342

Simon, D. (2006). "The continuous-time kalman filter," in *Optimal State Estimation,* ed D. Simon (Hoboken, NJ: John Wiley & Sons, Inc.), 229–262. doi: 10.1002/0470045345.ch8

Smith, D., and Singh, S. (2006). Approaches to multisensor data fusion in target tracking: a survey. *IEEE Trans. Knowl. Data Eng.* 18, 1696–1710. doi: 10.1109/TKDE.2006.183

Sze, V., Chen, Y.-H., Yang, T.-J., and Emer, J. S. (2017). Efficient processing of deep neural networks: a tutorial and survey. *Proc. IEEE* 105, 2295–2329. doi: 10.1109/JPROC.2017.2761740

Wan, W., Lu, F., Wu, Z., and Harada, K. (2017). Teaching robots to do object assembly using multi-modal 3D vision. *Neurocomputing* 259, 85–93. doi: 10.1016/j.neucom.2017.01.077

Wang, X., Zhang, T., Xu, X., Chen, L., Xing, X., and Chen, C. L. P. (2018). "EEG emotion recognition using dynamical graph convolutional neural networks and broad learning system," in *IEEE International Conference on Bioinformatics and Biomedicine* (Madrid), 1240–1244. doi: 10.1109/BIBM.2018.8621147

Waske, B., and Benediktsson, J. A. (2007). Fusion of support vector machines for classification of multisensor data. *IEEE Trans. Geosci. Remote Sens.* 45, 3858–3866. doi: 10.1109/TGRS.2007.898446

Yang, C., Chen, C., He, W., Cui, R., and Li, Z. (2018). Robot learning system based on adaptive neural control and dynamic movement primitives. *IEEE Trans. Neural Networks Learn. Syst.* 30, 777–787. doi: 10.1109/TNNLS.2018.2852711

Zeng, C., Yang, C., Zhong, J., and Zhang, J. (2019). Encoding multiple sensor data for robotic learning skills from multimodal demonstration. *IEEE Access* 7, 145604–145613. doi: 10.1109/ACCESS.2019.2945484

Zhang, T., Liu, Z., Wang, X., Xing, X., Chen, C. L. P., and Chen, E. (2018). "Facial expression recognition via broad learning system," in *IEEE International Conference on Systems, Man, and Cybernetics* (Miyazaki), 1898–1902. doi: 10.1109/SMC.2018.00328

Zhao, H., Zheng, J., Xu, J., and Deng, W. (2019). Fault diagnosis method based on principal component analysis and broad learning system. *IEEE Access* 7, 99263–99272. doi: 10.1109/ACCESS.2019.2929094

A Lyapunov-Stable Adaptive Method to Approximate Sensorimotor Models for Sensor-Based Control

David Navarro-Alarcon[1]*, Jiaming Qi[1], Jihong Zhu[2] and Andrea Cherubini[2]

[1] The Hong Kong Polytechnic University, Hong Kong, Hong Kong, [2] Université de Montpellier/LIRMM, Montpellier, France

In this article, we present a new scheme that approximates unknown sensorimotor models of robots by using feedback signals only. The formulation of the uncalibrated sensor-based regulation problem is first formulated, then, we develop a computational method that distributes the model estimation problem amongst multiple adaptive units that specialize in a local sensorimotor map. Different from traditional estimation algorithms, the proposed method requires little data to train and constrain it (the number of required data points can be analytically determined) and has rigorous stability properties (the conditions to satisfy Lyapunov stability are derived). Numerical simulations and experimental results are presented to validate the proposed method.

Keywords: robotics, sensorimotor models, adaptive systems, sensor-based control, servomechanisms, visual servoing

1. INTRODUCTION

Robots are widely used in industry to perform a myriad of sensor-based applications ranging from visually servoed pick-and-place tasks to force-regulated workpiece assemblies (Nof, 1999). Their accurate operation is largely due to the fact that industrial robots rely on fixed settings that enable the exact characterization of the tasks' sensorimotor model. Although this full characterization requirement is fairly acceptable in industrial environments, it is too stringent for many service applications where the mechanical, perceptual and environment conditions are not exactly known or might suddenly change (Navarro-Alarcon et al., 2019), e.g., in domestic robotics (where environments are highly dynamic), field robotics (where variable morphologies are needed to navigate complex workspaces), autonomous systems (where robots must adapt and operate after malfunctions), to name a few cases.

In contrast to industrial robots, the human brain has a high degree of adaptability that allows it to continuously learn sensorimotor relations. The brain can seemingly coordinate the body (whose morphology persistently changes throughout life) under multiple circumstances: severe injuries, amputations, manipulating tools, using prosthetics, etc. It can also recalibrate corrupted or modified perceptual systems: a classical example is the manipulation experiment performed in Kohler (1962) with image inverting goggles that altered a subject's visual system. In infants, motor babbling is used for obtaining (partly from scratch and partly innate) a coarse sensorimotor model that is gradually refined with repetitions (Von Hofsten, 1982). Providing robots with similar incremental and life-long adaptation capabilities is precisely our goal in this paper.

From an automatic control point of view, a sensorimotor model is needed for coordinating input motions of a mechanism with output sensor signals (Huang and Lin, 1994), e.g., controlling the shape of a manipulated soft object based on vision (Navarro-Alarcon et al., 2016) or controlling

the balance of a walking machine based on a gyroscope (Yu et al., 2018). In the visual servoing literature, the model is typically represented by the so-called interaction matrix (Hutchinson et al., 1996; Cherubini et al., 2015), which is computed based on kinematic relations between the robot's configuration and the camera's image projections. In the general case, sensorimotor models depend on the physics involved in constructing the output sensory signal; If this information is uncertain (e.g., due to bending of robot links, repositioning of external sensors, deformation of objects), the robot may no longer properly coordinate actions with perception. Therefore, it is important to develop methods that can efficiently provide robots with the capability to adapt to unforeseen changes of the sensorimotor conditions.

Classical methods in robotics to compute this model (see Sigaud et al., 2011 for a review) can be roughly classified into *structure-based* and *structure-free* approaches (Navarro-Alarcon et al., 2019). The former category represents "calibration-like" techniques [e.g., off-line (Wei et al., 1986) or adaptive (Wang et al., 2008; Liu et al., 2013; Navarro-Alarcon et al., 2015)] that aim to identify the unknown model parameters. These approaches are easy to implement, however, they require exact knowledge of the analytical structure of the sensory signal (which might not be available or subject to large uncertainties). Also, since the resulting model is fixed to the mechanical/perceptual/environmental setup that was used for computing it, these methods are not robust to unforeseen changes.

For the latter (structure-free) category, we can further distinguish between two main types (Navarro-Alarcon et al., 2019): instantaneous and distributed estimation. The first type performs online numerical approximations of the unknown model (whose structure does not need to be known); Some common implementations include e.g., Broyden-like methods (Hosoda and Asada, 1994; Jagersand et al., 1997; Alambeigi et al., 2018) and iterative gradient descent rules (Navarro-Alarcon et al., 2015; Yip et al., 2017). These methods are robust to sudden configuration changes, yet, as the sensorimotor mappings are continuously updated, they do not preserve knowledge of previous estimations (i.e., it's model is only valid for the current local configuration). The second type distributes the estimation problem amongst multiple computing units; The most common implementation is based on (highly nonlinear) connectionists architectures (Li and Cheah, 2014; Lyu and Cheah, 2018; Hu et al., 2019). These approaches require very large amounts of training data to properly constrain the learning algorithm, which is impractical in many situations. Other distributed implementations (based on SOM-like sensorimotor "patches," Kohonen, 2013) are reported e.g., in Zahra and Navarro-Alarcon (2019), Pierris and Dahl (2017), and Escobar-Juarez et al. (2016), yet, the stability properties of its algorithms are not rigorously analyzed.

As a solution to these issues, in this paper we propose a new approach that approximates unknown sensorimotor models based on local data observations only. In contrast to previous state-of-the-art methods, our adaptive algorithm has the following original features:

- It requires few data observations to train and constrain the algorithm (which allows to implement it in real-time).
- The number of minimum data points to train it can be analytically obtained (which makes data collection more effective).
- The stability of its update rule can be rigorously proved (which enables to deterministically predict its performance).

The proposed method is general enough to be used with different types of sensor signals and robot mechanisms.

The rest of the manuscript is organized as follows: section 2 presents preliminaries, section 3 describes the proposed method, section 4 reports the conducted numerical study, and section 5 gives final conclusions.

2. PRELIMINARIES

2.1. Notation
Along this note we use very standard notation. Column vectors are denoted with bold small letters $\mathbf{m}$ and matrices with bold capital letters $\mathbf{M}$. Time evolving variables are represented as $\mathbf{m}_t$, where the subscript $*_t$ denotes the discrete time instant. Gradients of functions $b = \beta(\mathbf{m}): \mathcal{M} \mapsto \mathcal{B}$ are denoted as $\nabla\beta(\mathbf{m}) = (\partial\beta/\partial\mathbf{m})^\mathsf{T}$.

2.2. Configuration Dependant Feedback
Consider a fully-actuated robotic system whose instantaneous configuration vector (modeling e.g., end-effector positions in a manipulator, orientation in a robot head, etc.) is denoted by the vector $\mathbf{x}_t \in \mathbb{R}^n$. Such model can only be used to represent traditional *rigid* systems, thus, it excludes soft/continuum mechanisms (Falkenhahn et al., 2015) or robots driven by elastic actuators (Wang et al., 2016). Without loss of generality, we assume that its coordinates are all represented using the same unitless range[1]. To perform a task, the robot is equipped with a sensing system that continuously measure a physical quantity whose instantaneous values depend on $\mathbf{x}_t$. Some examples of these types of configuration-dependent feedback signals are: geometric features in an image (Tirindelli et al., 2020), forces applied onto a compliant surface (Navarro-Alarcon et al., 2014; Bouyarmane et al., 2019), proximity to an object (Cherubini and Chaumette, 2013), intensity of an audio source (Magassouba et al., 2016), attitude of a balancing body (Defoort and Murakami, 2009), shape of a manipulated object (Navarro-Alarcon and Liu, 2018), temperature from a heat source (Saponaro et al., 2015), etc.

Let $\mathbf{y}_t \in \mathbb{R}^m$ denote the vector of feedback features that quantify the task; Its coordinates might be constructed with raw measurements or be the result of some processing. We model the instantaneous relation between this sensor signal and the robot's configuration as (Chaumette and Hutchinson, 2006):

$$\mathbf{y}_t = f(\mathbf{x}_t): \mathbb{R}^n \mapsto \mathbb{R}^m \qquad (1)$$

Remark 1. *Along this paper, we assume that the feedback feature functional $f(\mathbf{x}_t)$ is smooth (at least twice differentiable) and its*

---

[1]This can be easily obtained with constant kinematic transformations.

Jacobian matrix has a full row/column rank (which guarantees the existence of its (pseudo-)inverse).

2.3. Uncalibrated Sensorimotor Control

In our formulation of the problem, it is assumed that the robotic system is controlled via a standard position/velocity interface (as in e.g., Whitney, 1969; Siciliano, 1990), a situation that closely models the majority of commercial robots. With position interfaces, the motor action $\mathbf{u}_t \in \mathbb{R}^n$ represents the following displacement difference:

$$\mathbf{x}_{t+1} - \mathbf{x}_t = \mathbf{u}_t \tag{2}$$

Such *kinematic control* interface renders the typical stiff behavior present in industrial robots (for this model, external forces do not affect the robot's trajectories). The methods in this paper are formulated using position commands, however, these can be easily transformed into robot velocities $\mathbf{v}_t \in \mathbb{R}^n$ by dividing $\mathbf{u}_t$ by the servo controller's time step dt as follows $\mathbf{u}_t / dt = \mathbf{v}_t$.

The expression that describes how the motor actions result in changes of feedback features is represented by the first-order difference model[2]:

$$\mathbf{y}_{t+1} = \mathbf{y}_t + \mathbf{A}(\mathbf{x}_t)\mathbf{u}_t = \mathbf{y}_t + \boldsymbol{\delta}_t \tag{3}$$

where the configuration-dependent matrix $\mathbf{A}(\mathbf{x}_t) = \partial f / \partial \mathbf{x}_t \in \mathbb{R}^{m \times n}$ represents the traditional sensor Jacobian matrix of the system (also known as the interaction matrix in the visual servoing literature; Hutchinson et al., 1996). To simplify notation, throughout this paper we shall omit its dependency on $\mathbf{x}_t$ and denote it as $\mathbf{A}_t = \mathbf{A}(\mathbf{x}_t)$. The flow vector $\boldsymbol{\delta}_t = \mathbf{A}_t \mathbf{u}_t \in \mathbb{R}^m$ represents the sensor changes that result from the action $\mathbf{u}_t$. **Figure 1** conceptually depicts these quantities.

The sensorimotor control problem consists in computing the necessary motor actions for the robot to achieve a desired sensor configuration. Without loss of generality, in this note, such configuration is characterized as the regulation of the feature vector $\mathbf{y}_t$ toward a constant target $\mathbf{y}^*$. The necessary motor action to reach the target can be computed by minimizing the following quadratic cost function:

$$J = \left\| \lambda \, \text{sat}(\mathbf{y}_t - \mathbf{y}^*) + \mathbf{A}_t \mathbf{u}_t \right\|^2 \tag{4}$$

where $\lambda > 0$ is a gain and $\text{sat}(\cdot)$ a standard saturation function (defined as in e.g., Chang et al., 2018). The rationale behind the minimization of the cost (4) is to find an incremental motor command $\mathbf{u}_t$ that forward-projects into the sensory space (via the interaction matrix $\mathbf{A}_t$) as a vector pointing toward the target $\mathbf{y}^*$. By iteratively commanding these motions, the distance $\|\mathbf{y}_t - \mathbf{y}^*\|$ is expected to be asymptotically minimized.

To obtain $\mathbf{u}_t$, let us first compute the extremum $\nabla J(\mathbf{u}_t) = \mathbf{0}$, which yields the normal equation

$$\mathbf{A}_t^{\mathsf{T}} \mathbf{A}_t \mathbf{u}_t = -\lambda \mathbf{A}_t^{\mathsf{T}} \, \text{sat}(\mathbf{y}_t - \mathbf{y}^*) \tag{5}$$

---

[2]This difference equation represents the discrete-time model of the robot's differential sensor kinematics.
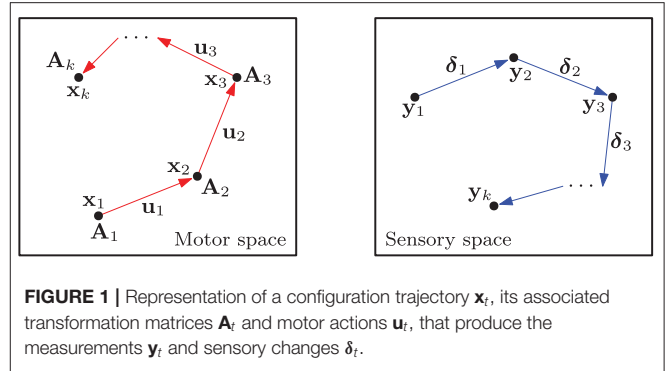
FIGURE 1 | Representation of a configuration trajectory $\mathbf{x}_t$, its associated transformation matrices $\mathbf{A}_t$ and motor actions $\mathbf{u}_t$, that produce the measurements $\mathbf{y}_t$ and sensory changes $\boldsymbol{\delta}_t$.

Solving (5) for $\mathbf{u}_t$, gives rise to the motor command that minimizes J:

$$\mathbf{u}_t = -\lambda \mathbf{A}_t^{\#} \, \text{sat}(\mathbf{y}_t - \mathbf{y}^*) \tag{6}$$

where $\mathbf{A}_t^{\#} \in \mathbb{R}^{n \times m}$ is a generalized pseudo-inverse matrix satisfying $\mathbf{A}_t \mathbf{A}_t^{\#} \mathbf{A}_t = \mathbf{A}_t$ (Nakamura, 1991), whose existence is guaranteed as $\mathbf{A}_t$ has a full column/row rank (depending on whichever is larger n or m). Yet, note that for the case where $m > n$, the cost function J can only be locally minimized.

Note that the computation of (6) requires exact knowledge of $\mathbf{A}_t$. To analytically calculate this matrix, we need to *fully calibrate* the system, which is too restrictive for applications where the sensorimotor model is unavailable or might suddenly change. This situation may happen if the mechanical structure of the robot is altered (e.g., due to bendings or damage of links), or the configuration of the perceptual system is changed (e.g., due to relocating external sensors), or the geometry of a manipulated object changes (e.g., due to grasping forces deforming a soft body), to name a few cases. Without this information, the robot may not properly coordinate actions with perception. In the following section, we describe our proposed solution.

3. METHODS

3.1. Discrete Configuration Space

Since the (generally non-linear) feature functional (1) is smooth, the Jacobian matrix $\mathbf{A}_t = \partial f / \partial \mathbf{x}_t$ is also expected to smoothly change along the robot's configuration space. This situation means that a *local* estimation $\widehat{\mathbf{A}}$ of the true matrix $\mathbf{A}_t$ around a configuration point $\mathbf{x}_i$ is also valid around the surrounding neighborhood (Sang and Tao, 2012). We exploit this simple yet powerful idea to develop a computational method that distributes the model estimation problem amongst various units that specialize in a local sensorimotor map.

It has been proved in the sensor-based control community (Cheah et al., 2003) that rough estimations of $\mathbf{A}_t$ (combined with the rectifying action of feedback) are sufficient for guiding the robot with sensory signals. However, note that large deviations from such configuration point $\mathbf{x}_i$ may result in model inaccuracies. Therefore, the local neighborhoods cannot be too large.

Consider a system with N computing units distributed around the robot's configuration space (see **Figure 2**). The location of these units can be defined with many approaches, e.g., with self-organization (Kohonen, 2001), random distributions, uniform distributions, etc. (Haykin, 2009). To each unit, we associate the following 3-tuple:

$$z^l = \left\{ \mathbf{w}^l \;\; \widehat{\mathbf{A}}_t^l \;\; \mathcal{D}^l \right\}, \quad \text{for} \quad l = 1, \dots, N \tag{7}$$

The weight vector $\mathbf{w}^l \in \mathbb{R}^n$ represents a configuration $\mathbf{x}_t$ of the robot where $\mathbf{w}^l = \mathbf{x}_t$. The matrix $\widehat{\mathbf{A}}_t^l \in \mathbb{R}^{n \times m}$ stands for a local approximation of $\mathbf{A}_t(\mathbf{w}^l)$ evaluated at the point $\mathbf{w}^l$. The purpose of the structure $\mathcal{D}^l$ is to store sensor and motor observations $\mathbf{d}_t = \{\mathbf{x}_t, \mathbf{u}_t, \boldsymbol{\delta}_t\}$, that are collected around the vicinity of $\mathbf{w}^l$ through babbling-like motions (Saegusa et al., 2009). The structure $\mathcal{D}^l$ is constructed as follows:

$$\mathcal{D}^l = \left\{ \mathbf{d}_1 \;\; \mathbf{d}_2 \;\; \cdots \;\; \mathbf{d}_\tau \right\}^\mathsf{T} \tag{8}$$

for $\tau > 0$ as the total number of observations, which once collected, they remain constant during the learning stage. Note that $\mathbf{x}_i$ and $\mathbf{x}_{i+1}$ are typically not consecutive time instances. The total number τ of observations is assumed to satisfy $\tau > mn$.

3.2. Initial Learning Stage

We propose an adaptive method to iteratively compute the local transformation matrix from data observations. To this end, consider the following quadratic cost function for the lth unit:

$$\begin{aligned} Q^l &= \frac{1}{2} \sum_{k=1}^\tau h^{lk} \left\| \widehat{\mathbf{A}}_t^l \mathbf{u}_k - \boldsymbol{\delta}_k \right\|^2 \\ &= \frac{1}{2} \sum_{k=1}^\tau h^{lk} \left\| \mathbf{F}(\mathbf{u}_k) \widehat{\mathbf{a}}_t^l - \boldsymbol{\delta}_k \right\|^2 \end{aligned} \tag{9}$$

for $\mathbf{F}(\mathbf{u}_k) \in \mathbb{R}^{m \times mn}$ as a regression-like matrix defined as

$$\mathbf{F}(\mathbf{u}_k) = \begin{bmatrix} \mathbf{u}_k^\mathsf{T} & \mathbf{0}_n^\mathsf{T} & \cdots & \mathbf{0}_n^\mathsf{T} \\ \mathbf{0}_n^\mathsf{T} & \mathbf{u}_k^\mathsf{T} & \cdots & \mathbf{0}_n^\mathsf{T} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0}_n^\mathsf{T} & \mathbf{0}_n^\mathsf{T} & \cdots & \mathbf{u}_k^\mathsf{T} \end{bmatrix} \tag{10}$$

FIGURE 2 | Representation of the lth computing unit and the neighboring data used to approximate the local sensorimotor model. The black and red dashed depict the Gaussian and its square approximation.
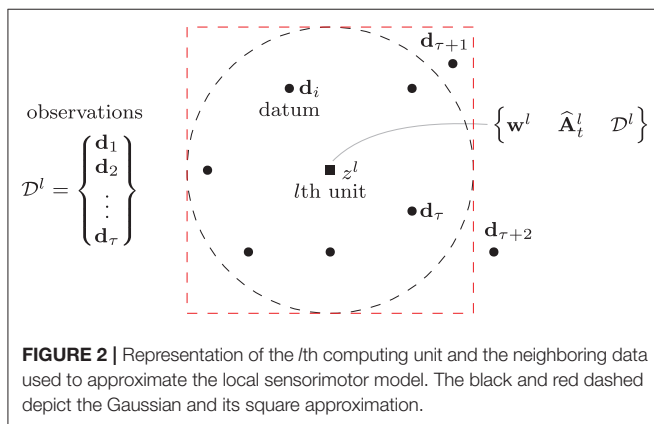
and a vector of adaptive parameters $\widehat{\mathbf{a}}_t^l \in \mathbb{R}^{nm}$ constructed as:

$$\widehat{\mathbf{a}}_t^l = \begin{bmatrix} \hat{a}_t^{l11} & \hat{a}_t^{l12} & \cdots & \hat{a}_t^{lmn} \end{bmatrix}^\mathsf{T} \tag{11}$$

where the scalar $\hat{a}_t^{lij}$ denotes the ith row jth column element of the matrix $\widehat{\mathbf{A}}_t^l$.

The scalar h^{lk} represents a Gaussian neighborhood function centered at the lth unit and computed as:

$$h^{lk} = \exp\left(-\frac{\|\mathbf{w}^l - \mathbf{x}_k\|^2}{2\sigma^2} \right) \tag{12}$$

where $\sigma > 0$ (representing the standard deviation) is used to control the width of the neighborhood. By using h^{lk}, the observations' contribution to the cost (9) proportionally decreases with the distance to $\mathbf{w}^l$. The dimension of the neighborhood is defined such that $h \approx 0$ is never satisfied for any of its observations $\mathbf{x}^k$. In practice, it is common to approximate the Gaussian shape with a simple "square" region, which presents the highest approximation error around its corners (see e.g., **Figure 2** where the sampling point $\mathbf{d}_{\tau+1}$ is within its boundary).

To compute an accurate sensorimotor model, the data points in (8) should be as distinctive as possible (i.e., the motor observations $\mathbf{u}_t$ should not be collinear). This requirement can be fairly achieved by covering the uncertain configuration with curved/random motions.

The following gradient descent rule is used for approximating the transformation matrix $\mathbf{A}_t$ at the lth unit:

$$\begin{aligned} \widehat{\mathbf{a}}_{t+1}^l &= \widehat{\mathbf{a}}_t^l - \gamma \nabla Q^l(\widehat{\mathbf{a}}_t^l) \\ &= \widehat{\mathbf{a}}_t^l - \gamma \sum_{k=1}^\tau h^{lk} \mathbf{F}(\mathbf{u}_k)^\mathsf{T} \left(\widehat{\mathbf{A}}_t^l \mathbf{u}_k - \boldsymbol{\delta}_k \right) \end{aligned} \tag{13}$$

for $\gamma > 0$ as a positive learning gain. For ease of implementation, the update rule (13) can be equivalently expressed in scalar form as:

$$\hat{a}_{t+1}^{lij} = \hat{a}_t^{lij} - \gamma \sum_{k=1}^\tau h^{lk} u_k^j \left\{ \left(\sum_{r=1}^n \hat{a}_t^{lir} u_k^r \right) - \delta_k^i \right\} \tag{14}$$

where u_k^j and δ_k^i denote the jth and ith components of the vectors $\mathbf{u}_k$ and $\boldsymbol{\delta}_k$, respectively.

Remark 2. *There are other estimation methods in the literature that also make use of Gaussian functions, e.g., radial basis functions (RBF) (Li and Cheah, 2014) to name an instance. However, RBF (in its standard formulation) use configuration-dependent Gaussians to modulate a set of weights (which provide non-linear approximation capabilities), whereas in our case, the Gaussians are used but within the weights' adaptation law to proportionally scale the contribution of the collected sensory-motor data (our method provides a* linear *approximation within the neighborhood). Our Gaussian weighted approach most closely resembles the one used in self organizing maps (SOM) (Kohonen, 2013) to combine surrounding data observations.*

3.3. Lyapunov Stability

In this section, we analyse the stability properties of the proposed update rule by using discrete-time Lyapunov theory (Bof et al., 2018). To this end, let us first assume that the transformation matrix satisfies:

$$\mathbf{A}(\mathbf{w}^l) = \partial f / \partial \mathbf{x}(\mathbf{w}^l) \approx \mathbf{A}(\mathbf{x}_j) \qquad (15)$$

for any configuration $\mathbf{x}_j$ around the neighborhood defined by $\mathcal{D}^l$ (this situation implies that $\mathbf{A}(\cdot)$ is constant around the vicinity of $\mathbf{w}^l$). Therefore, we can locally express around $\mathbf{w}^l$ the sensor changes as:

$$\delta_k = \mathbf{F}(\mathbf{u}_k)\mathbf{a}^l \qquad (16)$$

where $\mathbf{a}^l = [a^{l11}, a^{l12}, \dots, a^{lmn}]^\mathsf{T} \in \mathbb{R}^{mn}$ denotes the vector of *constant* parameters, for a^{lij} as the ith row jth column of the unknown matrix $\mathbf{A}(\mathbf{w}^l)$. To simplify notation, we shall denote $\mathbf{F}_k = \mathbf{F}(\mathbf{u}_k)$.

Proposition 1. *For a number mn of linearly independent vectors $\mathbf{u}_k$, the adaptive update rule (13) asymptotically minimizes the magnitude of the parameter estimation error $\|\widehat{\mathbf{a}}_t^l - \mathbf{a}^l\|$.*

Proof: Consider the following quadratic (energy-like) function:

$$V_t^l = \left\| \widehat{\mathbf{a}}_t^l - \mathbf{a}^l \right\|^2 \qquad (17)$$

Computing the forward difference of V_t^l yields:

$$
\begin{aligned}
V_{t+1}^l - V_t^l &= \left\| \widehat{\mathbf{a}}_{t+1}^l - \mathbf{a}^l \right\|^2 - \left\| \widehat{\mathbf{a}}_t^l - \mathbf{a}^l \right\|^2 \\
&= \left\| \left[\mathbf{I} - \gamma \sum_{k=1}^{\tau} h^{lk} \mathbf{F}_k^\mathsf{T} \mathbf{F}_k \right] \left(\widehat{\mathbf{a}}_t^l - \mathbf{a}^l \right) \right\|^2 \\
&\quad - \left\| \widehat{\mathbf{a}}_t^l - \mathbf{a}^l \right\|^2 = -\left(\widehat{\mathbf{a}}_t^l - \mathbf{a}^l \right)^\mathsf{T} \mathbf{\Omega} \left(\widehat{\mathbf{a}}_t^l - \mathbf{a}^l \right)
\end{aligned}
$$

for a symmetric matrix $\mathbf{\Omega} \in \mathbb{R}^{mn \times mn}$ defined as follows:

$$
\begin{aligned}
\mathbf{\Omega} &= \mathbf{I} - \left[\mathbf{I} - \gamma \sum_{k=1}^{\tau} h^{lk} \mathbf{F}_k^\mathsf{T} \mathbf{F}_k \right]^2 \\
&= 2\gamma \sum_{k=1}^{\tau} h^{lk} \mathbf{F}_k^\mathsf{T} \mathbf{F}_k - \gamma^2 \left[\sum_{k=1}^{\tau} h^{lk} \mathbf{F}_k^\mathsf{T} \mathbf{F}_k \right]^2 \\
&= \gamma \mathbf{\Phi}^\mathsf{T} \underbrace{\left(2\mathbf{H} - \gamma \mathbf{H} \mathbf{\Phi} \mathbf{\Phi}^\mathsf{T} \mathbf{H} \right)}_{\mathbf{C}} \mathbf{\Phi} \qquad (18)
\end{aligned}
$$

with $\mathbf{H} = \mathrm{diag}(h^{l1} \mathbf{I}_\tau, \dots, h^{l\tau} \mathbf{I}_\tau) \in \mathbb{R}^{m\tau \times m\tau}$ as a positive-definite diagonal matrix, $\mathbf{I}_\tau \in \mathbb{R}^{\tau \times \tau}$ as an identity matrix and $\mathbf{\Phi} \in \mathbb{R}^{m\tau \times mn}$ constructed with τ matrices $\mathbf{F}_k$ as follows:

$$\mathbf{\Phi} = \begin{bmatrix} \mathbf{F}_1^\mathsf{T} & \mathbf{F}_2^\mathsf{T} & \cdots & \mathbf{F}_\tau^\mathsf{T} \end{bmatrix}^\mathsf{T} \qquad (19)$$

To prove the asymptotic stability of (13), we must first prove the positive-definiteness of the dissipation-like matrix $\mathbf{\Omega}$ (van der

---

Algorithm 1: Compute a suitable γ.

1: $\gamma \leftarrow$ initial value < 1, $\mu \leftarrow$ small step
2: **repeat**
3: $\quad \gamma \leftarrow \gamma - \mu$
4: **until** $\mathbf{C} > \mathbf{0}$

Schaft, 2000). To this end, note that since the "tall" observations' matrix $\mathbf{\Phi}$ is exactly known and $\mathbf{H}$ is diagonal and positive (hence full-rank), we can always find a gain $\gamma > 0$ to guarantee that the symmetric matrix

$$\mathbf{C} = 2\mathbf{H} - \gamma \mathbf{H} \mathbf{\Phi} \mathbf{\Phi}^\mathsf{T} \mathbf{H} > 0, \qquad (20)$$

is also positive-definite, and therefore, full-rank. Next, let us re-arrange mn linearly *independent* row vectors from $\mathbf{\Phi}$ as follows:

$$
\begin{bmatrix}
\mathbf{u}_1^\mathsf{T} & \mathbf{0}_n^\mathsf{T} & \cdots & \mathbf{0}_n^\mathsf{T} \\
\mathbf{u}_2^\mathsf{T} & \mathbf{0}_n^\mathsf{T} & \cdots & \mathbf{0}_n^\mathsf{T} \\
\vdots & \vdots & \vdots & \vdots \\
\mathbf{u}_n^\mathsf{T} & \mathbf{0}_n^\mathsf{T} & \cdots & \mathbf{0}_n^\mathsf{T} \\
\mathbf{0}_n^\mathsf{T} & \mathbf{u}_{n+1}^\mathsf{T} & \cdots & \mathbf{0}_n^\mathsf{T} \\
\mathbf{0}_n^\mathsf{T} & \mathbf{u}_{n+2}^\mathsf{T} & \cdots & \mathbf{0}_n^\mathsf{T} \\
\vdots & \vdots & \ddots & \vdots \\
\mathbf{0}_n^\mathsf{T} & \mathbf{0}_n^\mathsf{T} & \cdots & \mathbf{u}_{mn-1}^\mathsf{T} \\
\mathbf{0}_n^\mathsf{T} & \mathbf{0}_n^\mathsf{T} & \cdots & \mathbf{u}_{mn}^\mathsf{T}
\end{bmatrix} \qquad (21)
$$

which shows that $\mathbf{\Phi}$ has a full column rank, hence, the matrix $\mathbf{\Omega} = \gamma \mathbf{\Phi}^\mathsf{T} \mathbf{C} \mathbf{\Phi} > 0$ is positive-definite. This condition implies that $V_{t+1}^l - V_t^l < 0$ for any $\|\widehat{\mathbf{a}}_t^l - \mathbf{a}^l\| \neq 0$. Asymptotic stability of the parameter's estimation error directly follows by invoking Lyapunov's direct method (Bof et al., 2018). ∎

Remark 3. *There are two conditions that need to be satisfied to ensure the algorithm's stability. The first condition is related to the magnitude of the learning gain γ. Large gain values may lead to numerical instabilities, which is a common situation in discrete-time adaptive systems. To find a "small enough" gain $\gamma > 0$, we can conduct the simple 1D search shown in Algorithm 1. An eigenvalue test on $\mathbf{C}$ can be used to verify (20). The second condition is related to the linear independence (i.e., the non-collinearity) of the motor actions $\mathbf{u}_t$. Such independent vectors are needed for providing a sufficient number of constraints to the estimation algorithm (this condition can be easily satisfied by performing random babbling-like motions).*

3.4. Localized Adaptation

Once the cost function (9) has been minimized, the computed transformation matrix $\widehat{\mathbf{A}}_t$ *locally approximates* the robot's sensorimotor model around the lth unit. Note that the stability of the total N units is analogous the analysis shown in the previous section; A global analysis is out of the scope of this work.

The associated local training data (8) must then be released from memory to allow for new relations to be learnt—if needed. However, for the case where changes in the sensorimotor conditions occur, the model may contain inaccuracies in some or

all computing units, and thus, its transformation matrices cannot be used for controlling the robot's motion. To cope with this issue, we need to first quantitatively assess such errors. For that, the following weighted *distortion* metric is introduced:

$$U_t = \mathbf{e}_t^\mathsf{T} \mathbf{B} \mathbf{e}_t \tag{22}$$

where $\mathbf{B} > 0$ denotes a positive-definite diagonal weight matrix to homogenize different scales in the approximation error $\mathbf{e}_t = \widehat{\mathbf{A}}^s \mathbf{u}_t - \boldsymbol{\delta}_t \in \mathbb{R}^m$. The scalar index s is found by solving the search problem:

$$s = \arg \min_j \|\mathbf{w}^j - \mathbf{x}_t\| \tag{23}$$

To enable adaptation of problematic units, we evaluate the magnitude of the metric U_t, and if found to be larger than an arbitrary threshold $U_t > |\varepsilon|$, new motion and sensor data must be collected around the sth computing unit to construct the revised structure $\mathcal{D}^s$ by using a *push* approach:

$$\mathbf{d}_1 \leftarrow \{\mathbf{x}_t \ \mathbf{u}_t \ \boldsymbol{\delta}_t\} \tag{24}$$

that updates the topmost observation and discards the oldest (bottom) data, so as to keep a constant number τ of data points. The transformation matrices are then computed with the new data.

3.5. Motion Controller

The update rule (13) computes an adaptive transformation matrix $\widehat{\mathbf{A}}_t^l$ for each of the N units in the system. To provide a smooth transition between different units, let us introduce the matrix $\mathbf{L}_t \in \mathbb{R}^{m \times n}$ which is updated as follows[3]:

$$\mathbf{L}_{t+1} = \mathbf{L}_t - \eta \left(\mathbf{L}_t - \widehat{\mathbf{A}}_t^s \right) \tag{25}$$

where $\eta > 0$ is a tuning gain. The above matrix represents a filtered version of $\widehat{\mathbf{A}}_t^s$, where s denotes the index of the active unit, as defined in (23). With this approach, the transformation matrix smoothly changes between adjacent neighborhoods, while providing stable values in the vicinity of the active unit; It can be seen as a continuous interpolation between adjacent neighborhoods.

The motor command with adaptive model is implemented as follows:

$$\mathbf{u}_t = -\lambda \mathbf{L}_t^{\#} \mathrm{sat}(\mathbf{y}_t - \mathbf{y}^*) \tag{26}$$

The stability of this *kinematic control* method can be analyzed with its resulting closed-loop first-order system (a practice also commonly adopted with visual servoing controllers; Chaumette and Hutchinson, 2006). To this end, we use a small displacement approach (motivated by the local target provided by the saturation function), where we introduce the increment vector $\mathbf{i} = -\mathrm{sat}(\mathbf{y}_t - \mathbf{y}^*)$ and define the local reference position $\overline{\mathbf{y}} = \mathbf{y}_t + \mathbf{i} \in \mathbb{R}^m$. Let us consider the case when the N units have minimized the cost functions (9). Note that the asymptotic

---

[3]For simplicity, we initialize $\mathbf{L}_0 = \mathbf{0}_{n \times n}$ with a zero matrix.

minimization of $\|\widehat{\mathbf{a}}_t^l - \mathbf{a}^l\|$ implies that $\widehat{\mathbf{A}}_t^s$ inherits the rank properties of $\mathbf{A}_t$, hence, the existence of the pseudo-inverse in (26) is guaranteed; A regularization term (see e.g., Tikhonov et al., 2013) can further be used to robustify the computation of $\mathbf{L}_t^{\#}$.

Proposition 2. *For $n \geq m$ (i.e., more/equal motor actions than feedback features), the "stiff" kinematic control input (26) provides the local feedback error $\mathbf{y}_t - \overline{\mathbf{y}}$ with asymptotic stability.*

Proof: Substitution of the controller (26) into the difference model (3) yields the closed-loop system:

$$\begin{aligned} \mathbf{y}_{t+1} &= \mathbf{y}_t - \lambda \, \mathrm{sat}(\mathbf{y}_t - \mathbf{y}^*) = \mathbf{y}_t + \lambda \mathbf{i} \pm \lambda \mathbf{y}_t \\ &= \mathbf{y}_t - \lambda \mathbf{y}_t + \lambda \overline{\mathbf{y}} = \mathbf{y}_t - \lambda (\mathbf{y}_t - \overline{\mathbf{y}}) \end{aligned} \tag{27}$$

Adding $\pm \overline{\mathbf{y}}$ to (27) and after some algebraic operation, we obtain:

$$\left(\mathbf{y}_{t+1} - \overline{\mathbf{y}} \right) = (1 - \lambda) \left(\mathbf{y}_t - \overline{\mathbf{y}} \right) \tag{28}$$

which for a gain satisfying $0 < \lambda < 1$, it implies local asymptotic stability of the small displacement error $(\mathbf{y}_t - \overline{\mathbf{y}})$ (Kuo, 1992).

Remark 4. *Note that the above stability analysis assumes that robot's trajectories are not perturbed by external forces and that the estimated interaction matrix locally satisfies $\mathbf{A}_t \mathbf{L}_t^{\#} \mathbf{A}_t \approx \mathbf{A}_t$ around the active neighborhood.*

4. CASE OF STUDY

In this section, we validate the performance of the proposed method with numerical simulations and experiments. A vision-based manipulation task with a deformable cable is used as our case of study (Bretl and McCarthy, 2014): It consists in the robot actively deforming the object into a desired shape by using visual feedback of the cable's contour (see e.g., Zhu et al., 2018). Soft object manipulation tasks are challenging—and relevant to the fundamental problem addressed here—since the sensorimotor models of deformable objects are typically unknown or subject to large uncertainties (Sanchez et al., 2018). Therefore, the transformation matrix relating the shape feature functional and the robot motions is difficult to compute. The proposed algorithm will be used to adaptively approximate the unknown model. **Figure 3** conceptually depicts the setup of this sensorimotor control problem.

4.1. Simulation Setup

For this study, we consider a planar robot arm that rigidly grasps one end of an elastic cable, whose other end is static; We assume that the total motion of this composed cable-robot system remains on the plane. A monocular vision sensor observes the manipulated cable and measures its 2D contour in real-time. The dynamic behavior of the elastic cable is simulated as in Wakamatsu and Hirai (2004) by using the minimum energy principle (Hamill, 2014), whose solution is computed using the CasADi framework (Andersson et al., 2019). The cable is assumed to have negligible plastic behavior. All numerical simulation algorithms are implemented in MATLAB.

FIGURE 3 | Representation of the cable manipulation case of study, where a vision sensor continuously measures the cable's feedback shape $\mathbf{y}_t$, which must be actively deformed toward $\mathbf{y}*$.

FIGURE 4 | Various configurations of the visually measured cable profile (black solid line) and its approximation with Fourier series (red dashed line).
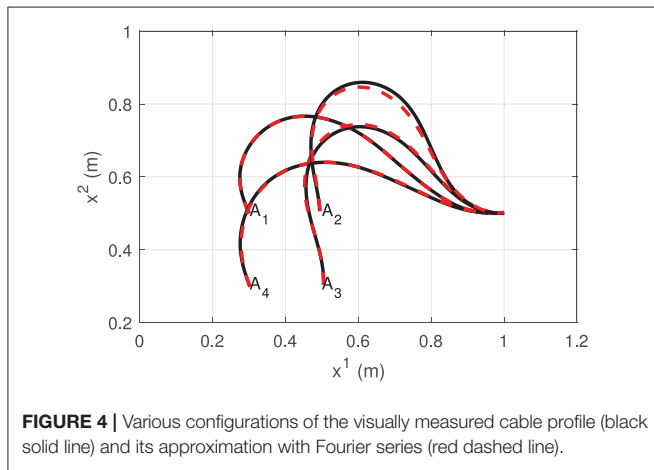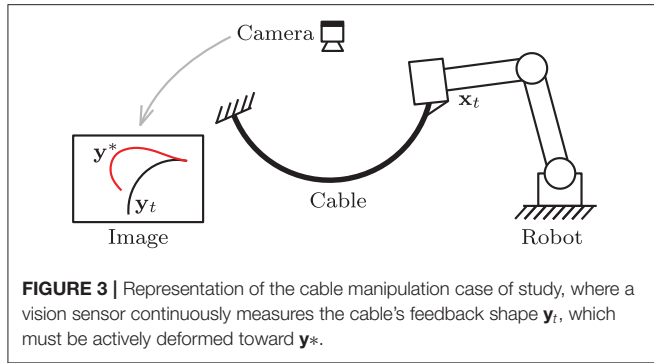
The cable simulation code is publicly available at https://github.com/Jihong-Zhu/cableModelling2D.

Let the long vector $\mathbf{s}_t \in \mathbb{R}^{2\alpha}$ represents the 2D profile of the cable, which is simulated using a resolution of $\alpha = 100$ data points. To perform the task, we must compute a vector of feedback features $\mathbf{y}_t$ that characterizes the object's configuration. For that, we use the approach described in Digumarti et al. (2019) and Navarro-Alarcon and Liu (2018) that approximates $\mathbf{s}_t$ with truncated Fourier series (in our case, we used four harmonics), and then constructs $\mathbf{y}_t$ with the respective Fourier coefficients (Collewet and Chaumette, 2000). The use of these coefficients as feedback signals enable us to obtain a compact representation of the object's configuration, however, it complicates the analytical derivation of the matrix $\mathbf{A}_t$.

4.2. Approximation of the Matrix $\mathbf{A}_t$

To construct the data structure (8), we collect $\tau = 40$ data observations $\mathbf{d}_t$ at random locations around the manipulation workspace. Next, we define local neighborhoods centered at the configuration points $\mathbf{w}^1 = [0.3, 0.5]$, $\mathbf{w}^2 = [0.5, 0.5]$, $\mathbf{w}^3 = [0.5, 0.3]$, and $\mathbf{w}^4 = [0.5, 0.5]$. These neighborhoods are defined with a standard deviation of $\sigma = 1.3$. With the collected observations, $l = 1, \ldots, 4$ matrices $\widehat{\mathbf{A}}_t^l$ are computed using the update rule (14).

Figure 4 depicts the measured shape (black solid line) of the cable at the four points $\mathbf{w}^l$ and the shape that is approximated (red

FIGURE 5 | Profile of the function G that is computed along the circular trajectory passing through the points in **Figure 4**; The "switch" label indicates the instant when $\widehat{\mathbf{A}}_t^l$ switches to different one.
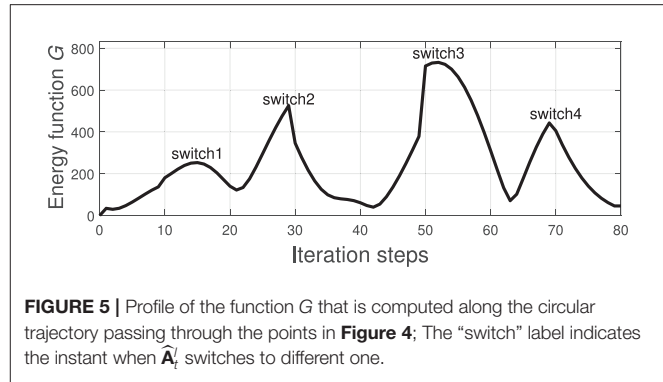
dashed line) with the feedback feature vector $\mathbf{y}_t$ (i.e the Fourier coefficients). It shows that four harmonics provide sufficient accuracy for representing the object's configuration. To evaluate the accuracy of the computed discrete configuration space and its associated matrices $\widehat{\mathbf{A}}_t^l$, we conduct the following test: The robot is commanded to move the cable along a circular trajectory that passes through the four points $\mathbf{w}^l$. The following energy function is computed throughout this trajectory:

$$G = \left\| \boldsymbol{\delta}_t - \widehat{\mathbf{A}}_t^l \mathbf{u}_t \right\|^2 \tag{29}$$

which quantifies the accuracy of the local differential mapping (3). The index l switches (based on the solution of 23) as the robot enters a different neighborhood.

Figure 5 depicts the profile of the function G along the trajectory. We can see that this error function increases as the robot approaches the neighborhood's boundary. The "switch" label indicates the time instant when $\mathbf{A}_t^l$ switches to different (more accurate) matrix, an action that decreases the magnitude of G. This result confirms that the proposed adaptive algorithm provides local directional information on how the motor actions transform into sensor changes.

4.3. Sensor-Guided Motion

In this section, we make use of the approximated sensorimotor model to guide the motion of a robotic system based on feedback features. To this end, various cable shapes are defined as target configurations $\mathbf{y}^*$ (to provide physically feasible targets, these shapes are collected from previous sensor observations). The target configurations are then given to the motion controller (26) to automatically perform the task. The controller implemented with saturation bounds of $|\operatorname{sat}(\cdot)| \leq 2$ and a feedback gain $\lambda = 0.1$.

Figure 6 depicts the progression of the cable shapes obtained during these numerical simulations. The initial $\mathbf{y}_0$ and the intermediate configurations are represented with solid black curves, whereas the final shape $\mathbf{y}^*$ is represented with red dashed curves. To assess the accuracy of the controller, the following cost function is computed throughout the shaping motions:

$$E = \left\| \mathbf{y}_t - \mathbf{y}^* \right\|^2 \tag{30}$$

For these four shaping actions, **Figure 7** depicts the time evolution of the function E. This figure clearly shows that the feedback error is asymptotically minimized.

Now, consider the setup depicted in **Figure 8**, which has two 3-DOF robots jointly manipulating the deformable cable. For this more complex scenario, the total configuration vector $\mathbf{x}_t$ must be constructed with the 3-DOF pose (position and orientation) vectors of both robot manipulators as $\mathbf{x}_t =$

$[{}^{L}\mathbf{x}_t^{\mathsf{T}}, {}^{R}\mathbf{x}_t^{\mathsf{T}}]^{\mathsf{T}} \in \mathbb{R}^6$. Training of the sensorimotor model is done similarly as with the single-robot case described above; The same feedback gains and controller parameters are also used in this test.

Figure 9 depicts the initial shape $\mathbf{y}_0$ and intermediate configurations (black solid curves), as well as the respective final shape $\mathbf{y}^*$ (red dashed curve) of the cable. Note that as more input DOF can be controlled by the robotic system, the object can be actively deformed into more complex configurations (cf. the achieved S-shape curve with the profiles in **Figure 6**). The result demonstrates that the approximated sensorimotor model provides sufficient directional information to the controller to properly "steer" the feature vector $\mathbf{y}_t$ toward the target $\mathbf{y}^*$.

FIGURE 6 | Initial and final configurations of four different shape control simulations **(A–D)**, using a single robot manipulator.
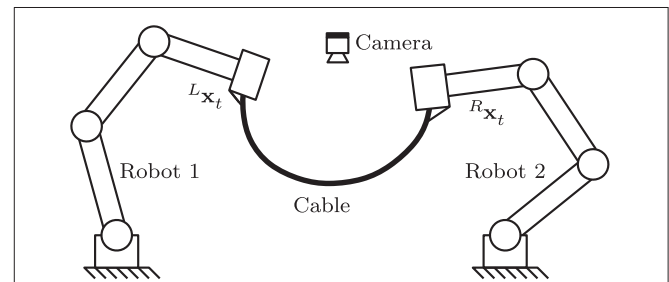
FIGURE 8 | Representation of a two-robot setup where both systems must jointly shape the cable into a desired form.
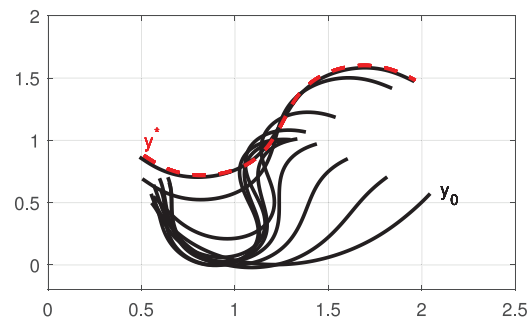
FIGURE 9 | Initial and final configurations of the shape control simulation with two robots.

FIGURE 7 | Minimization process of the energy function E.
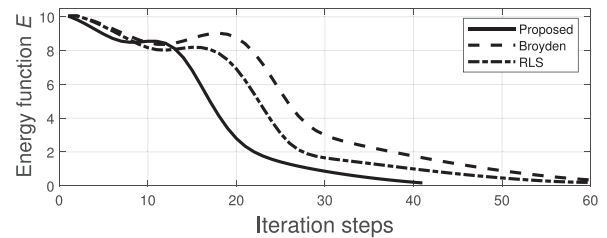
FIGURE 10 | Minimization process of the energy function E.

We now compare the performance of our method (using the same manipulation task shown in **Figures 8**, **9**) with two state-of-the-art approaches commonly used for guiding robots with unknown sensorimotor models. To this end, we consider the classical Broyden update rule (Broyden, 1965) and the recursive least-squares (RLS) (Hosoda and Asada, 1994). These two methods are used for estimating the matrix **A** that is needed to compute the control input (6). To compare their performance,

FIGURE 13 | Asymptotic minimization of the error functional E obtained with the experiments shown in **Figure 12**.

FIGURE 11 | The experimental robotic setup.
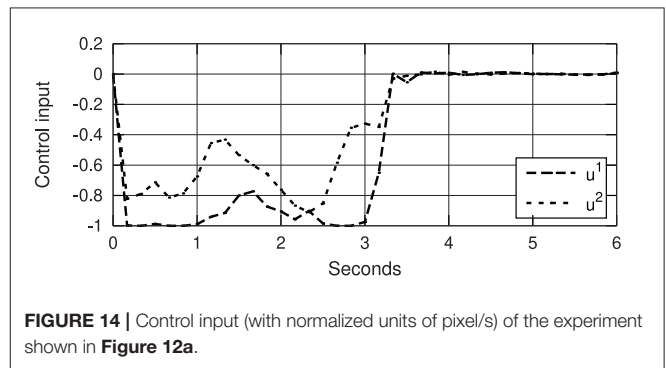
FIGURE 14 | Control input (with normalized units of pixel/s) of the experiment shown in **Figure 12a**.
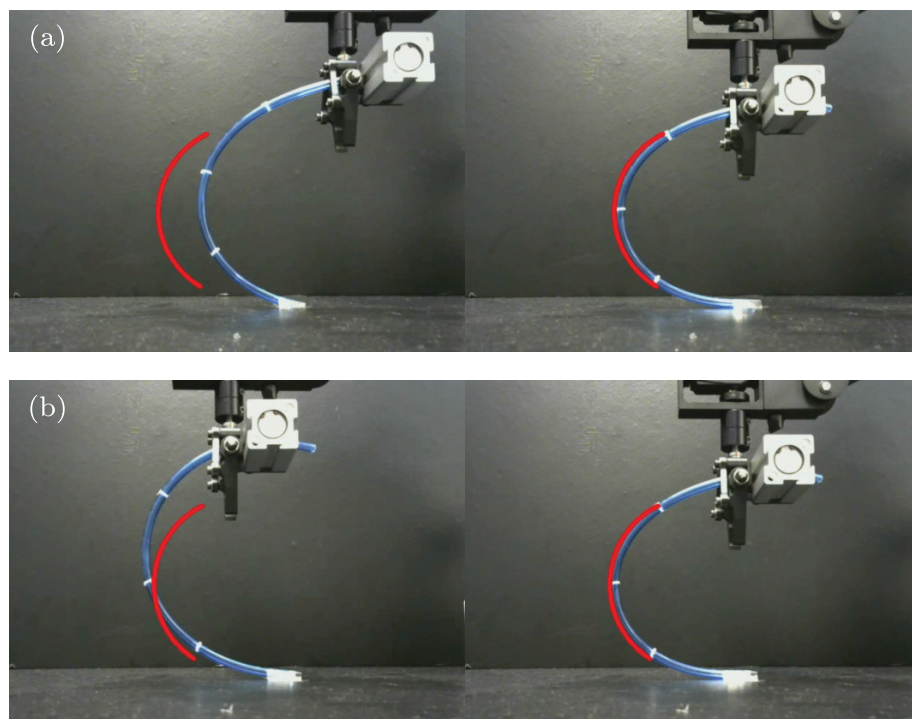
FIGURE 12 | Snapshots of the initial (left) and final (right) configurations for two shape control experiments **(a)** and **(b)**, where the red curve represents the target shape.

FIGURE 15 | Control input (with normalized units of pixel/s) of the experiment shown in **Figure 12b**.
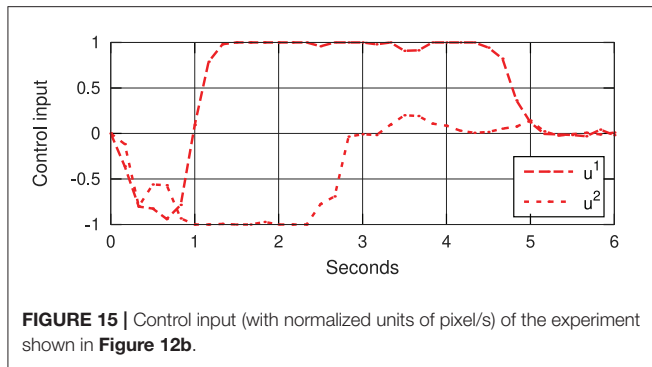
the cost function E is evaluated throughout their respective trajectories; The same feedback gain $\lambda = 0.1$ is used for these three methods. **Figure 10** depicts the time evolution of E computed with the three methods. This result demonstrates that the performance of our method is comparable to the other two classical approaches.

4.4. Experiments

To validate the proposed theory, we developed an experimental platform composed of a three degrees-of-freedom serial robotic manipulator (DOBOT Magician), a Linux-based motion control system (Ubuntu 16.04), and a USB Webcam (Logitech C270); Image processing is performed by using the OpenCV libraries (Bradski, 2000). A sampling time of $dt \approx 0.04$ s is used in our Linux-based control system. In this setup, the robot rigidly grasps an elastic piece of pneumatic air tubing, whose other end is attached to the ground. The 3-DOF mechanism has a double parallelogram structure that enables to control the gripper's x-y-z position while keeping a constant orientation. For this experimental study, we only control 2-DOF of the robot such it manipulates the tubing with plane motions. **Figure 11** depicts the setup.

We conduct similar vision-guided experiments with the platform as the ones described in the previous section. For these tasks, the elastic tubing must be automatically positioned into a desired contour. The configuration dependant feedback for this task is computed with the observed contour of the object by using two harmonic terms (Navarro-Alarcon and Liu, 2018). The sensorimotor model is similarly approximated around four configuration points (as in **Figure 4**), by performing random motions and collecting sensor data.

Figure 12 depicts snapshots of the conducted experiments, where we can see the initial and final configurations of the system. The red curves represent the (static) target configuration $\mathbf{y}^*$. For these two targets, **Figure 13** depicts the respective time evolution profiles of the energy function E, where we can clearly see that the feedback error is asymptotically minimized. The control inputs $\mathbf{u}_t$ used during the experiments are depicted in **Figures 14**, **15**. These motion commands are computed from raw vision measurements and a saturation threshold of ± 1 is applied to its values. This results demonstrate that the approximated model can be used to locally guide motions of the robot with sensor feedback.

5. CONCLUSION

In this paper, we describe a method to estimate sensorimotor relations of robotic systems. For that, we present a novel adaptive rule that computes local sensorimotor relations in real-time; The stability of this algorithm is rigorously analyzed and its convergence conditions are derived. A motion controller to coordinate sensor measurements and robot motions is proposed. Simulation and experimental results with a cable manipulation case of study are reported to validate the theory.

The main idea behind the proposed method is to divide the robot's configuration workspace into discrete nodes, and then, locally approximate at each node the mappings between robot motions and sensor changes. This approach resembles the estimation of piecewise linear systems, except that in our case, the computed model represents a differential Jacobian-like relation. The key to guarantee the stability of the algorithm lies in collecting sufficient linear independent motor actions (such condition can be achieved by performing random babbling motions).

The main limitation of the proposed algorithm is the local nature of its model, which can be improved by increasing the density of the distributed computing units. Another issue is related to the scalability of its discretized configuration space. Note that for 3D spaces, the method can fairly well approximate the sensorimotor model, yet for multiple DOF (e.g., more than 6) the data is difficult to manage and visualize.

As future work, we would like to implement our adaptive method with other sensing modalities and mechanical configurations, e.g., with an eye-in-hand visual servoing (where the camera orientation is arbitrary) and with variable morphology manipulators (where the link's length and joint's configuration are not known).

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author/s.

AUTHOR CONTRIBUTIONS

DN-A conceived the algorithm and drafted the manuscript. JQ and JZ performed the numerical simulation results. AC analyzed the theory and revised the paper. All authors contributed to the article and approved the submitted version.

FUNDING

REFERENCES

Alambeigi, F., Wang, Z., Hegeman, R., Liu, Y., and Armand, M. (2018). A robust data-driven approach for online learning and manipulation of unmodeled 3-d heterogeneous compliant objects. *IEEE Robot. Autom. Lett.* 3, 4140–4147. doi: 10.1109/LRA.2018.2863376

Andersson, J. A. E., Gillis, J., Horn, G., Rawlings, J. B., and Diehl, M. (2019). CasADi-A software framework for nonlinear optimization and optimal control. *Math. Prog. Comp.* 11, 1–36. doi: 10.1007/s12532-018-0139-4

Bof, N., Carli, R., and Schenato, L. (2018). Lyapunov theory for discrete time systems. *CoRR abs/1809.05289*. Available online at: https://arxiv.org/abs/1809.05289

Bouyarmane, K., Chappellet, K., Vaillant, J., and Kheddar, A. (2019). Quadratic programming for multirobot and task-space force control. *IEEE Trans. Robot.* 35, 64–77. doi: 10.1109/TRO.2018.2876782

Bradski, G. (2000). The OpenCV library. *Dr. Dobb's J. Softw. Tools* 25, 122–125.

Bretl, T., and McCarthy, Z. (2014). Quasi-static manipulation of a Kirchhoff elastic rod based on a geometric analysis of equilibrium configurations. *Int. J. Robot. Res.* 33, 48–68. doi: 10.1177/0278364912473169

Broyden, C. G. (1965). A class of methods for solving nonlinear simultaneous equations. *Math. Comp.* 19, 577–593. doi: 10.1090/S0025-5718-1965-0198670-6

Chang, H., Wang, S., and Sun, P. (2018). Omniwheel touchdown characteristics and adaptive saturated control for a human support robot. *IEEE Access* 6, 51174–51186. doi: 10.1109/ACCESS.2018.2869836

Chaumette, F., and Hutchinson, S. (2006). Visual servo control. Part I: Basic approaches. *IEEE Robot. Autom. Mag.* 13, 82–90. doi: 10.1109/MRA.2006.250573

Cheah, C. C., Hirano, M., Kawamura, S., and Arimoto, S. (2003). Approximate Jacobian control for robots with uncertain kinematics and dynamics. *IEEE Trans. Robot. Autom.* 19, 692–702. doi: 10.1109/TRA.2003.814517

Cherubini, A., and Chaumette, F. (2013). Visual navigation of a mobile robot with laser-based collision avoidance. *Int. J. Robot. Res.* 32, 189–205. doi: 10.1177/0278364912460413

Cherubini, A., Passama, R., Fraisse, P., and Crosnier, A. (2015). A unified multimodal control framework for human-robot interaction. *Robot. Auton. Syst.* 70, 106–115. doi: 10.1016/j.robot.2015.03.002

Collewet, C., and Chaumette, F. (2000). "A contour approach for image-based control on objects with complex shape," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, Vol. 1* (Takamatsu), 751–756. doi: 10.1109/IROS.2000.894694

Defoort, M., and Murakami, T. (2009). Sliding-mode control scheme for an intelligent bicycle. *IEEE Trans. Indus. Electron.* 56, 3357–3368. doi: 10.1109/TIE.2009.2017096

Digumarti, K. M., Trimmer, B., Conn, A. T., and Rossiter, J. (2019). Quantifying dynamic shapes in soft morphologies. *Soft Robot.* 1, 1–12. doi: 10.1089/soro.2018.0105

Escobar-Juarez, E., Schillaci, G., Hermosillo-Valadez, J., and Lara-Guzman, B. (2016). A self-organized internal models architecture for coding sensory–motor schemes. *Front. Robot. AI* 3:22. doi: 10.3389/frobt.2016.00022

Falkenhahn, V., Mahl, T., Hildebrandt, A., Neumann, R., and Sawodny, O. (2015). Dynamic modeling of bellows-actuated continuum robots using the euler-lagrange formalism. *IEEE Trans. Robot.* 31, 1483–1496. doi: 10.1109/TRO.2015.2496826

Hamill, P. (2014). *A Student's Guide to Lagrangians and Hamiltonians*. New York, NY: Cambridge University Press.

Haykin, S. (2009). *Neural Networks and Learning Machines*. Ontario: Prentice Hall.

Hosoda, K., and Asada, M. (1994). "Versatile visual servoing without knowledge of true Jacobian," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, Vol. 1* (Munich), 186–193. doi: 10.1109/IROS.1994.407392

Hu, Z., Han, T., Sun, P., Pan, J., and Manocha, D. (2019). 3-D deformable object manipulation using deep neural networks. *IEEE Robot. Autom. Lett.* 4, 4255–4261. doi: 10.1109/LRA.2019.2930476

Huang, J., and Lin, C.-F. (1994). On a robust nonlinear servomechanism problem. *IEEE Trans. Autom. Control* 39, 1510–1513. doi: 10.1109/9.299646

Hutchinson, S., Hager, G., and Corke, P. (1996). A tutorial on visual servo control. *IEEE Trans. Robot. Autom.* 12, 651–670. doi: 10.1109/70.538972

Jagersand, M., Fuentes, O., and Nelson, R. (1997). "Experimental evaluation of uncalibrated visual servoing for precision manipulation," in *Proceedings of IEEE International Conference on Robotics and Automation, Vol. 4* (Albuquerque, NM), 2874–2880. doi: 10.1109/ROBOT.1997.606723

Kohler, I. (1962). Experiments with goggles. *Sci. Am.* 206, 62–73. doi: 10.1038/scientificamerican0562-62

Kohonen, T. (2001). *Self-Organizing Maps*. Berlin; Heidelberg: Springer. doi: 10.1007/978-3-642-56927-2

Kohonen, T. (2013). Essentials of the self-organizing map. *Neural Netw.* 37, 52–65. doi: 10.1016/j.neunet.2012.09.018

Kuo, B. (1992). *Digital Control Systems*. Electrical Engineering. New York, NY: Oxford University Press.

Li, X., and Cheah, C. C. (2014). Adaptive neural network control of robot based on a unified objective bound. *IEEE Trans. Control Syst. Technol.* 22, 1032–1043. doi: 10.1109/TCST.2013.2293498

Liu, Y.-H., Wang, H., Chen, W., and Zhou, D. (2013). Adaptive visual servoing using common image features with unknown geometric parameters. *Automatica* 49, 2453–2460. doi: 10.1016/j.automatica.2013.04.018

Lyu, S., and Cheah, C. C. (2018). "Vision based neural network control of robot manipulators with unknown sensory Jacobian matrix," in *IEEE/ASME International Conference on Advanced Intelligent Mechatronics* (Auckland), 1222–1227. doi: 10.1109/AIM.2018.8452467

Magassouba, A., Bertin, N., and Chaumette, F. (2016). "Audio-based robot control from interchannel level difference and absolute sound energy," in *Proceedings of IEEE International Conference on Intelligent Robots and Systems* (Daejeon), 1992–1999. doi: 10.1109/IROS.2016.7759314

Nakamura, Y. (1991). *Advanced Robotics: Redundancy and Optimization*. Boston, MA: Addison-Wesley Longman.

Navarro-Alarcon, D., Cherubini, A., and Li, X. (2019). "On model adaptation for sensorimotor control of robots," in *Chinese Control Conference* (Guangzhou), 2548–2552. doi: 10.23919/ChiCC.2019.8865825

Navarro-Alarcon, D., and Liu, Y.-H. (2018). Fourier-based shape servoing: a new feedback method to actively deform soft objects into desired 2D image shapes. *IEEE Trans. Robot.* 34, 272–279. doi: 10.1109/TRO.2017.2765333

Navarro-Alarcon, D., Liu, Y.-H., Romero, J. G., and Li, P. (2014). Energy shaping methods for asymptotic force regulation of compliant mechanical systems. *IEEE Trans. Control Syst. Technol.* 22, 2376–2383. doi: 10.1109/TCST.2014.2309659

Navarro-Alarcon, D., Yip, H., Wang, Z., Liu, Y.-H., Zhong, F., Zhang, T., et al. (2016). Automatic 3D manipulation of soft objects by robotic arms with adaptive deformation model. *IEEE Trans. Robot.* 32, 429–441. doi: 10.1109/TRO.2016.2533639

Navarro-Alarcon, D., Yip, H. M., Wang, Z., Liu, Y.-H., Lin, W., and Li, P. (2015). "Adaptive image-based positioning of RCM mechanisms using angle and distance features," in *IEEE/RSJ International Conference on Intelligent Robots and Systems* (Hamburg), 5403–5409. doi: 10.1109/IROS.2015.7354141

Nof, S. (1999). *Handbook of Industrial Robotics*. New Jersey, NJ: John Wiley & Sons. doi: 10.1002/9780470172506

Pierris, G., and Dahl, T. S. (2017). Learning robot control using a hierarchical som-based encoding. *IEEE Trans. Cogn. Dev. Syst.* 9, 30–43. doi: 10.1109/TCDS.2017.2657744

Saegusa, R., Metta, G., Sandini, G., and Sakka, S. (2009). "Active motor babbling for sensorimotor learning," in *International Conference on Robotics and Biomimetics* (Bangkok), 794–799. doi: 10.1109/ROBIO.2009.4913101

Sanchez, J., Corrales, J.-A., Bouzgarrou, B.-C., and Mezouar, Y. (2018). Robotic manipulation and sensing of deformable objects in domestic and industrial applications: a survey. *Int. J. Robot. Res.* 37, 688–716. doi: 10.1177/0278364918779698

Sang, Q., and Tao, G. (2012). Adaptive control of piecewise linear systems: the state tracking case. *IEEE Trans. Autom. Control* 57, 522–528. doi: 10.1109/TAC.2011.2164738

Saponaro, P., Sorensen, S., Kolagunda, A., and Kambhamettu, C. (2015). "Material classification with thermal imagery," in *IEEE Conference on Computer Vision and Pattern Recognition* (Boston, MA), 4649–4656. doi: 10.1109/CVPR.2015.7299096

Siciliano, B. (1990). Kinematic control of redundant robot manipulators: a tutorial. *J. Intell. Robot. Syst.* 3, 201–212. doi: 10.1007/BF00126069

Sigaud, O., Salaün, C., and Padois, V. (2011). On-line regression algorithms for learning mechanical models of robots: a survey. *Rob. Auton. Syst.* 59, 1115–1129. doi: 10.1016/j.robot.2011.07.006

Tikhonov, A., Goncharsky, A., Stepanov, V., and Yagola, A. (2013). *Numerical Methods for the Solution of Ill-Posed Problems. Mathematics and Its Applications.* Dordrecht: Springer Netherlands.

Tirindelli, M., Victorova, M., Esteban, J., Kim, S. T., Navarro-Alarcon, D., Zheng, Y. P., et al. (2020). Force-ultrasound fusion: Bringing spine robotic-us to the next "level". *IEEE Robot. Autom. Lett.* 5, 5661–5668. doi: 10.1109/LRA.2020.3009069

van der Schaft, A. (2000). L_2-*Gain and Passivity Techniques in Nonlinear Control.* London: Springer. doi: 10.1007/978-1-4471-0507-7

Von Hofsten, C. (1982). Eye-hand coordination in the newborn. *Dev. Psychol.* 18:450. doi: 10.1037/0012-1649.18.3.450

Wakamatsu, H. and Hirai, S. (2004). Static modeling of linear object deformation based on differential geometry. *Int. J. Robot. Res.* 23, 293–311. doi: 10.1177/0278364904041882

Wang, H., Liu, Y.-H., and Zhou, D. (2008). Adaptive visual servoing using point and line features with an uncalibrated eye-in-hand camera. *IEEE Trans. Robot.* 24, 843–857. doi: 10.1109/TRO.2008.2001356

Wang, Z., Yip, H. M., Navarro-Alarcon, D., Li, P., Liu, Y., Sun, D., et al. (2016). Design of a novel compliant safe robot joint with multiple working states. *IEEE/ASME Trans. Mechatron.* 21, 1193–1198. doi: 10.1109/TMECH.2015.2500602

Wei, G.-Q., Arbter, K., and Hirzinger, G. (1986). Active self-calibration of robotic eyes and hand-eye relationships with model identification. *IEEE Trans. Robot. Autom.* 14, 158–166. doi: 10.1109/70.660864

Whitney, D. (1969). Resolved motion rate control of manipulators and human prostheses. *IEEE Trans. Man-Mach. Syst.* 10, 47–53. doi: 10.1109/TMMS.1969.299896

Yip, H. M., Navarro-Alarcon, D., and Liu, Y. (2017). "An image-based uterus positioning interface using adaline networks for robot-assisted hysterectomy," in *IEEE International Conference on Real-time Computing and Robotics* (Okinawa), 182–187. doi: 10.1109/RCAR.2017.8311857

Yu, C., Zhou, L., Qian, H., and Xu, Y. (2018). "Posture correction of quadruped robot for adaptive slope walking," in *IEEE International Conference on Robotics and Biomimetics* (Malaysia), 1220–1225. doi: 10.1109/ROBIO.2018.8665093

Zahra, O. and Navarro-Alarcon, D. (2019). "A self-organizing network with varying density structure for characterizing sensorimotor transformations in robotic systems," in *Towards Autonomous Robotic Systems*, eds K. Althoefer, J. Konstantinova, K. Zhang (London: Springer), 167–178. doi: 10.1007/978-3-030-25332-5_15

Zhu, J., Navarro, B., Fraisse, P., Crosnier, A., and Cherubini, A. (2018). "Dual-arm robotic manipulation of flexible cables," in *IEEE/RSJ Int. Conf. on Robots and Intelligent Systems* (Madrid), 479–484. doi: 10.1109/IROS.2018.8593780

Gait Neural Network for Human-Exoskeleton Interaction

Bin Fang[1], Quan Zhou[2], Fuchun Sun[1*], Jianhua Shan[2*], Ming Wang[3], Cheng Xiang[4] and Qin Zhang[5]

[1] Department of Computer Science and Technology, Beijing National Research Center for Information Science and Technology, Tsinghua University, Beijing, China, [2] Anhui Province Key Laboratory of Special Heavy Load Robot, Anhui University of Technology, Ma'anshan, China, [3] North Automatic Control Technology Institute, Taiyuan, China, [4] Department of Physics & Astronomy, Iowa State University, Ames, IA, United States, [5] State Key Lab of Digital Manufacturing Equipment and Technology, Huazhong University of Science and Technology, Wuhan, China

Robotic exoskeletons are developed with the aim of enhancing convenience and physical possibilities in daily life. However, at present, these devices lack sufficient synchronization with human movements. To optimize human-exoskeleton interaction, this article proposes a gait recognition and prediction model, called the gait neural network (GNN), which is based on the temporal convolutional network. It consists of an intermediate network, a target network, and a recognition and prediction model. The novel structure of the algorithm can make full use of the historical information from sensors. The performance of the GNN is evaluated based on the publicly available HuGaDB dataset, as well as on data collected by an inertial-based wearable motion capture device. The results show that the proposed approach is highly effective and achieves superior performance compared with existing methods.

1. INTRODUCTION

The development of lower-extremity robotic exoskeletons (Ackermann and van den Bogert, 2010) has been found to have significant potential in medical rehabilitation (Zhang et al., 2015) and military equipment applications. In these devices, human gait is captured in real time through signals (Casale et al., 2011) that are then sent to a controller. The controller returns instructions to the mechanical device on necessary adjustments or modifications. However, these exoskeletons require more effective prediction modules for joint gait trajectories (Aertbeliën and Schutter, 2014). The main goal is to improve the synchronization between the exoskeleton and human movement (Du et al., 2015). Essentially, the time gap between human action and mechanical device adjustment must be reduced without sacrificing the precision and quality of the modification. To achieve this, it is necessary to mine historical data and understand their intent (Zhu et al., 2016). Time series analysis is a powerful tool for this purpose.

Gait signal collection is commonly performed via inertial measurement units (IMUs), tactile sensors, surface electromyography, electroencephalograms, and so on. The majority of popular gait datasets employ computer vision technology to improve efficiency (Shotton et al., 2011), such as the Carnegie Mellon University Motion of Body dataset (Gross and Shi, 2001), the University of Maryland Human Identification at a Distance dataset, the Chinese Academy of Sciences Institute of Automation Gait Database (Zheng et al., 2011), and the Osaka University Institute of Science and Industry Research Gait Database (Iwama et al., 2012). However, it is often difficult to obtain accurate human gait information from such image-based prediction methods.

The HuGaDB dataset of the University Higher School of Economics contains highly detailed kinematic data for human gait analysis and activity recognition. It is the first public human gait dataset derived from inertial sensors that contains segmented annotations for the study of movement transitions. The data were obtained from 18 participants, and a total of about 10 h of activity was recorded (Chereshnev and Kertesz-Farkas, 2017). However, one limitation of the above-mentioned datasets is that they are not adequate for extremely delicate exoskeleton control.

To achieve sufficiently dexterous and adaptive control, in addition to statistical approximation by Markov modeling, deep learning has been demonstrated to be an effective approach. Recurrent neural networks, such as long short-term memory (LSTM), have been widely used in the areas of time series analysis and natural language processing. The cyclic nature of the human gait has previously precluded the use of such networks (Martinez et al., 2017; Ferrari et al., 2019). Nevertheless, a novel LSTM-based framework has been proposed for predicting the gait stability of elderly users of an intelligent robotic rollator (Chalvatzaki et al., 2018), fusing multimodal RGB-D and laser rangefinder data from non-wearable sensors (Chalvatzaki et al., 2018). An LSTM network has also been used to model gait synchronization of legs using a basic off-the-shelf IMU configuration with six acceleration and rotation parameters (Romero-Hernandez et al., 2019). Further, recent works have reported the use of convolutional neural networks (CNNs) for human activity recognition. CNNs use accelerometer data for real-time human activity recognition and can handle the extraction of both local features and simple statistical features that preserve information about the global form of a time series (Casale et al., 2011). A survey on deep learning for sensor-based activity recognition is presented in Wang et al. (2018).

At present, the majority of gait prediction models are not sufficiently precise or robust with respect to environmental fluctuations. In this work, a gait neural network (GNN) is proposed for gait recognition and prediction through wearable devices. The data-processing component consists of two phases: handling buffer data through an intermediate network and target prediction. Experiments are performed on a public human gait dataset, and the results obtained from the GNN are longitudinally compared with those of other methods. Further, more meticulous gait signals are collected from the IMU to improve training convergence and accuracy. Our model should also be helpful to exoskeletons with inertial sensors.

2. MATERIALS AND METHODS

CNNs are often used for two-dimensional data-processing tasks, such as image classification and target detection. Recently, researchers have found that CNNs can be used to process one-dimensional time series data, where both the convolution kernel and the pooling window are changed from having two dimensions to having just one dimension. In 2018, a temporal convolutional network (TCN) architecture was proposed (Bai et al., 2018). This architecture was deliberately kept simple, combining some of the best practices of modern convolutional architectures. When compared with canonical recurrent architectures, such as LSTM and gated recurrent units, the TCN can convincingly outperform baseline recurrent architectures across a broad range of sequence-modeling tasks. Some scholars have used TCNs in human action segmentation with video or image data (Lea et al., 2016a,b) and medical time series classification (Lin et al., 2019). The distinguishing characteristics of TCNs are that the convolutions in the architecture are causal, meaning that there is no information "leakage" from future to past, and that the architecture can take a sequence of any length and map it to an output sequence of the same length, just as with a recurrent neural network. Therefore, the TCN is used as the base model to handle sequence-modeling tasks, such as obtaining inertial data on human gaits.

2.1. Standard Temporal Convolutional Networks

Consider an input sequence $x_0, \ldots, x_T$ for which an output prediction, such as $y_0, \ldots, y_T$ is desired for each time, where y_T depends only on $x_0, \ldots, x_T$, with no future inputs $x_{t+1}, \ldots, x_T$. The sequence-modeling network is any function $f : x^{T+1} \rightarrow y^{T+1}$ that produces the mapping

$$\hat{y}_0, \ldots, \hat{y}_T = f(x_0, \ldots, x_T). \qquad (1)$$

This sequence-learning algorithm seeks a network f that minimizes the loss $L(y_0, \ldots, y_T, f(x_0, \ldots, x_T))$, which measures the difference between the predictions and the actual targets. The TCN employs dilated convolutions to allow an exponentially large receptive field. For a one-dimensional sequence input $x \in \mathbb{R}^n$ and a filter $f : \{0, \ldots, k-1\} \rightarrow \mathbb{R}$, the dilated convolution operation on an element of the sequence is defined as

$$F(s) = \sum_{i=0}^{k-1} f(i) \cdot x_{s-d \cdot i}, \qquad (2)$$

where d is the dilation factor, k is the filter size, and $s - d \cdot i$ accounts for the past direction. Thus, dilation is equivalent to introducing a fixed step between every two adjacent filter taps. When $d = 1$, a dilated convolution reduces to a regular convolution. By using a larger dilation, the outputs in the top layer can have a larger receptive field and so represent a wider range of inputs. The basic architectural elements of a TCN are shown in **Figure 1**.

The output o of a layer is related to the input via an activation function:

$$o = \text{Activation}(x + \Gamma(x)). \qquad (3)$$

Within a residual block, the base model has two layers of dilated causal convolution and non-linearity. For the non-linearity, a rectified linear unit is used as the activation function. For normalization, weight normalization is applied to the convolutional filters. For regularization, a spatial dropout is added after each dilated convolution. In a standard residual network the input is directly added to the output of the residual function, whereas in a TCN the input and output can have

FIGURE 1 | Structure of the GNN.
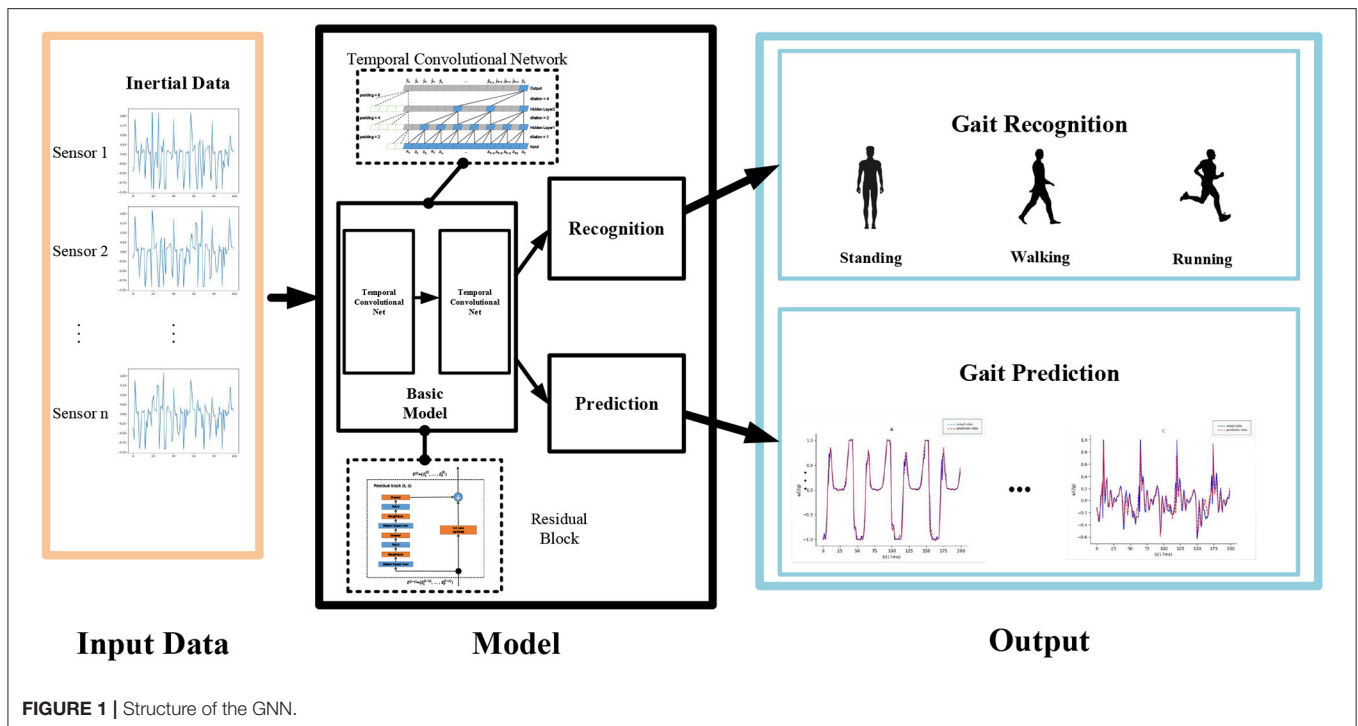
different widths. To account for possibly different input and output widths, an additional 1×1 convolution is used, which ensures that the elementwise addition receives tensors of the same shape.

For convenience, the GNN predicts only the acceleration and gyro data.

Gait prediction is a type of time series prediction. Because it has been shown that TCNs can be very effective sequence models for sequence data (MatthewDavies and Bock, 2019), a TCN is used for human gait analysis in the present work.

A TCN has two characteristics: dilated convolution and causal convolution. The primary function of the dilated convolution is to enable the network to learn more information in a long time series. However, it has been observed that long time series information does not significantly improve the accuracy of gait prediction, because human gait data are periodic and excess information is sampled repeatedly. Therefore, we disregard dilated convolution in this study.

2.2. Gait Neural Network

The architecture of the GNN is shown in **Figure 1**.

Two TCNs are used for the basic model. The first is the intermediate network, which uses the normalized inertial data as input to predict the intermediate sensor data. Unlike traditional methods, because of the response delay in the system, the GNN reserves some buffer time, and the original, buffer, and target data are represented, respectively by x, y, and z. Traditional methods generally use x to predict z, and y remains unused. As shown in **Figures 2**, **3**, the GNN uses the original signals as the intermediate network input x to predict the intermediate data y. The second TCN in the model is the target vector network. The

original data x and the predicted data y are concatenated to make the input to the target vector network, which then outputs the encoded vectors.

Firstly, one TCN is used to process the original input, and then the original input and the output of the first TCN are combined into the input to the second TCN. To a certain extent, it can be seen that the number of features of model learning is increased and the ability to obtain historical information is enhanced.

Finally, a recognition model and a prediction model are added to the network as two fully connected layers. The encoded vectors obtained from the basic model are fed into these recognition and prediction models to output the human action (walking, standing, or running) and the predicted gait data z (which mainly includes acceleration and gyro data).

2.2.1. Loss Function

A loss function is used to evaluate the fitting effect of a deep neural network. It is also used to compute gradients using a back-propagation algorithm to optimize the parameters of the network. The GNN has two loss functions to calculate: one for measuring the prediction loss due to prediction error and the other for measuring the loss of recognition accuracy.

The prediction loss function of the GNN is

$$L_{\text{pred}} = w_y * L_y(\hat{y}, y) + w_z * L_z(\hat{z}, z), \tag{4}$$

where L_y and L_z represent the loss functions of the intermediate prediction network and the target vector network, respectively; $\hat{z}$ and z are the output vector and the target vector, respectively; and w_y and w_z are the weight coefficients of L_y and L_z, respectively, for

FIGURE 2 | Composition of the basic GNN model.

FIGURE 3 | Input data of the GNN.
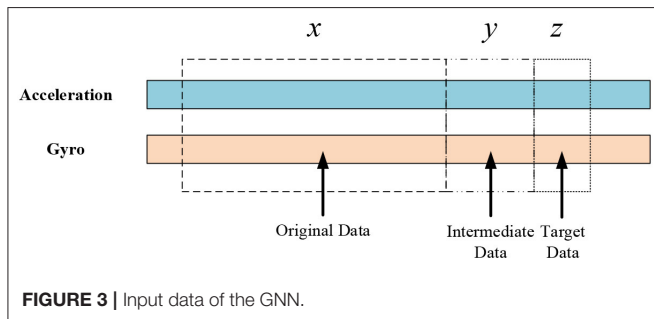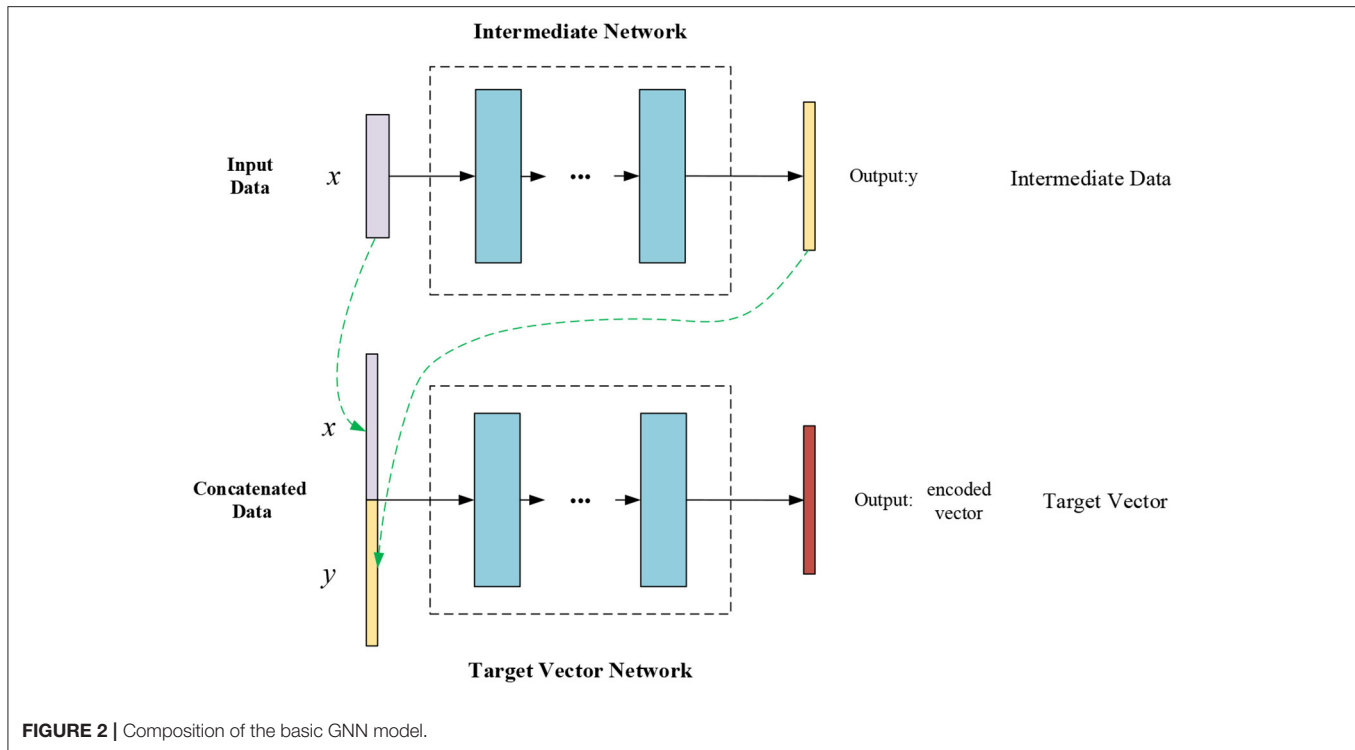
which either L_1 or L_2 loss functions can be used:

$$L_1 = \frac{1}{N_B} \sum_{k=1}^{N_B} \sum_{j} |\hat{u}_k^j - u_k^j|, \qquad (5)$$

$$L_2 = \frac{1}{2N_B} \sum_{k=1}^{N_B} \sum_{j} (\hat{u}_k^j - u_k^j)^2, \qquad (6)$$

where N_B denotes the batch size, which is in the range of {32, 64, 128, 256}, $\hat{u}$ is the predicted value of the network output, u is the tag value of the network output, and j indicates the jth output value of the network.

The recognition loss function of the GNN represents the cross-entropy loss:

$$L_{rec} = -\frac{1}{n} \sum [y \ln \hat{y} + (1 - y) \ln(1 - \hat{y})], \qquad (7)$$

where $\hat{y}$ is the model output.

The total loss of the GNN is

$$L_{total} = L_{pred} + \alpha L_{rec}, \qquad (8)$$

where α is the hyperparameter used to balance the loss function in order to achieve high performance during the recognition and prediction tasks.

2.3. Experimental Approach

In this study, the performance of the GNN was evaluated on two datasets: the publicly available HuGaDB dataset and a human gait dataset obtained using an inertial-based wearable motion capture device.

The GNN was trained by an Adam optimizer with a learning rate of 0.001 at 80 and 150 epochs, divided by 10. The maximum epoch and batch size were 200 and 64, respectively. The dropout rate of all dropout layers was set to 0.3. The GNN was implemented by PyTorch and trained and tested on a computer with an Intel Core i7-8750H processor, two 8 GB memory chips (DDR4), and a GPU (GeForce GTX 1060 6G).

2.4. Experiment on a Public Dataset
2.4.1. Gait Data Description

The human body has more than 200 bones. To simplify the gait analysis process, motion analysis is often performed on collected joint motion data. Gait analysis is one method used to study an individual's walking pattern. It aims to reveal the key links and factors influencing an abnormal gait through biomechanics and kinematics, in order to aid clinical diagnosis, guide rehabilitation evaluation and treatment, evaluate efficacy,

and inform research on the mechanisms involved. In gait analysis, special parameters are used to assess whether the gait is normal; these generally include gait cycle, kinematic, dynamic, myoelectric activity, and energy metabolism parameters. To improve the correspondence between the robotic exoskeleton and the human body, motion data, such as joint velocity and acceleration data must be collected.

Gait prediction in lower-extremity exoskeleton robots requires highly accurate human gait data, so it is necessary to utilize a gait dataset suitable for human gait prediction. Therefore, we chose to evaluate the gait prediction and recognition performance of the GNN on the publicly available HuGaDB dataset (Chereshnev and Kertesz-Farkas, 2017), which contains detailed kinematic data for analyzing human activity patterns, such as walking, running, taking the stairs up and down, and sitting down. The recorded data are segmented and annotated. They were obtained from a body sensor network comprising six wearable inertial sensors for collecting gait data. Sensors were placed on the left and right thighs, lower legs, and insteps of the human body; their distribution is shown in **Figure 4**. Each inertial sensor was used to collect three-axis accelerometer, three-axis gyroscope, and occasionally electromyography signals of the corresponding body part, providing data that can be used to evaluate the posture and joint angle of the lower limbs. The data were recorded from 18 participants, and consist of 598 min and 2,111,962 samples in total. The microcontroller collected 56.3500 samples per second on average, with a standard deviation of 3.2057, and transmitted them to a laptop through a Bluetooth connection (Chereshnev and Kertesz-Farkas, 2017). Only the inertial data were taken as input to the present model.

2.4.2. Data Analysis

In this work we analyze gait data from wearable sensors. Given the similarity between the two legs, we use only the right leg as an example. As seen in **Figure 5**, there is a certain relationship between the sensor signals; hence it is preferable to conduct training using all of the data as input rather than some subset. The acceleration signals of three people were randomly sampled for

data analysis. As shown in **Figure 6**, the gait data are periodic, and the patterns of the three people are similar, although the different individuals have different walking gaits.

2.4.3. Data Pre-processing

The original data in HuGaDB can be converted to values of the corresponding variables. To properly utilize the variables in the GNN, data normalization is necessary. The normalization formula is

$$x_{\text{norm}} = \frac{x - \text{mean}(x)}{\max(x)}. \tag{9}$$

By this formula, all gait data can be scaled to be between -1 and 1, which can eliminate learning difficulties caused by inconsistencies in data size and range.

2.4.4. Sample Creation

After preprocessing the gait data, the samples used for network training were created. The gait data consist of the acceleration and angular velocity of the inertial devices. In the experiment we conducted, the lengths of x, y, and z were 10, 5, and 1, respectively. To confirm the method of sample creation, samples were selected from the HuGaDB gait data. The first 80% of samples were used as the training dataset, the next 10% were selected for the validation dataset, and the last 10% were taken to be the test dataset.

2.5. Experiment on the Collected Data

To further evaluate the GNN, an inertial-based wearable motion capture device was used to collect human gait data. The entire motion acquisition system consists of seven inertial measurement units, and only the signals from the lower limbs were selected for the human gait prediction, as shown in **Figure 7**. Each unit

FIGURE 4 | Location of the sensors on the human body that collected data for HuGaDB.

FIGURE 5 | Correlation between the HuGaDB sensor data.
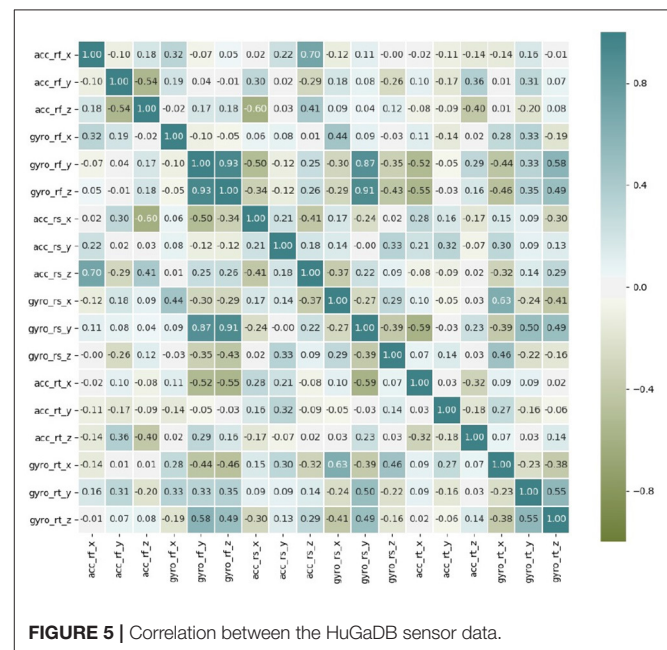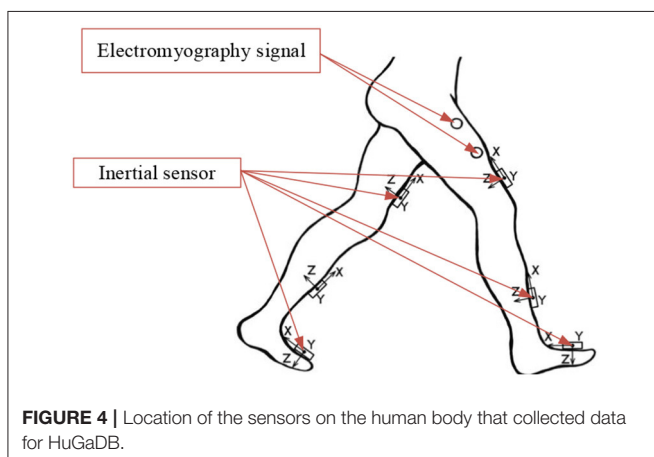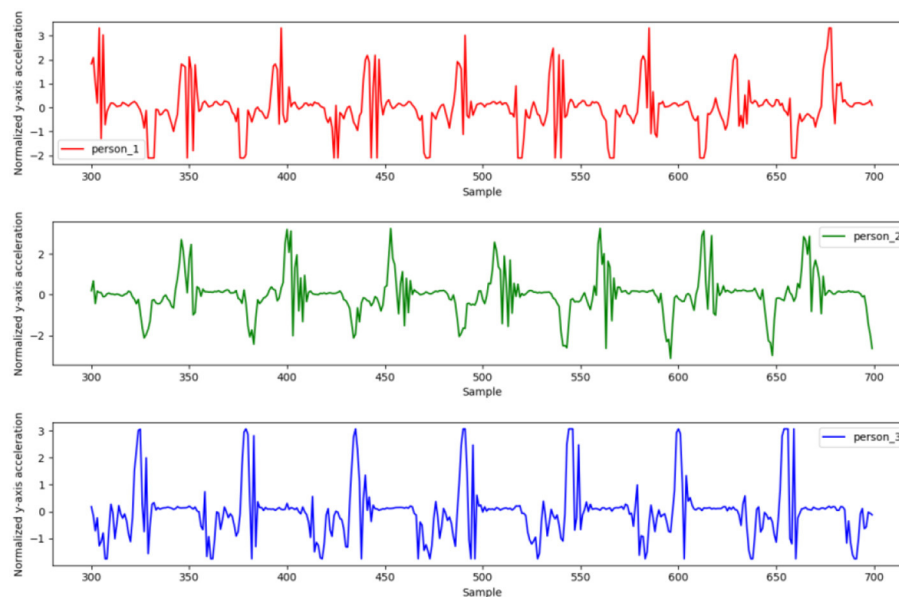
FIGURE 6 | Similarity in the acceleration data of three people.

FIGURE 7 | Collected human gait data.

measured the three-axis acceleration and the three-axis angular velocity. The sampling frequency was 120 Hz. The collected gait data include data on walking, going up and down stairs, and going up and down slopes.

3. RESULTS

As the predicted data can be collected by inertial sensors, the model is used to predict the accelerations and angular velocities. In the training of the model, multimodal data are used, and the numerical distributions in different dimensions are different; but since we need to use these data as input at the same time, we normalize the data in each dimension. The input and output of the model are multidimensional data, and the units in different

dimensions are different. The training of the model is based on the normalized data, and the predicted value of the model is also the accurate value after normalization. Therefore, the mean squared error (MSE) and mean absolute error (MAE) in the evaluation indexes are based on the normalized data, so the units are not specified.

3.1. Evaluation Results Using HuGaDB

The network was first trained using the gait data from a single wearer, and the results were compared with those obtained from existing methods. Considering that the exoskeleton must adapt to the movement of different wearers, to test the network's generalization ability, the gait data of three wearers were selected for the training set, and the data of one wearer who was not

included in the training set were used for the test set. It took 2.405 s to complete the recognition and prediction task on 2,249 samples in the test set, which means that the prediction and recognition task on each sample took only 1 ms.

3.1.1. Gait Prediction

The results in **Table 1** show that the GNN achieves the best results in the prediction task, with the exception of the maximum error for one wearer. Compared with the other methods, the GNN has the best generalization ability for new human gait data. Further, when the hyperparameter α is set to 0.4 in the experiments, the GNN shows the top performance. As displayed in **Figure 8**, the gait data of one wearer, including x-, y-, and z-axis acceleration data, were selected to make a prediction; the horizontal parts of the curve represent standing posture, while the oscillating parts represent the walking and running states. We find that the GNN produced good results.

3.1.2. Gait Recognition

As shown in **Table 2**, for the single wearer's motion data, all methods achieved good recognition results. When the GNN that was trained on three wearers' data received a new wearer's gait data as input, although it did not achieve the best performance, it did yield a promising accuracy rate of 98.04%.

Based on its performance in the human gait prediction and recognition tasks, we can conclude that the GNN is highly effective in the analysis of human motion.

3.2. Evaluation Results Using the Collected Data

To ensure the reliability and fairness of the experiment, all parameters of the model are the same as those used for the HuGaDB dataset.

3.2.1. Gait Prediction

As shown in **Figure 9**, the acceleration signal of the left tibia was selected as the prediction object. It can be seen that the GNN achieves good prediction performance, except for some abnormal points, which could be caused by noise. It took 1.2332 s to complete the recognition and prediction task on 674 samples in the test set, which means that the prediction and recognition task on each sample took only 1.8 ms.

3.2.2. Gait Recognition

The GNN was compared with other methods on the collected dataset; the results are shown in **Tables 3**, **4**. Clearly, the GNN achieved the best results in most tasks. Further, the MSE value is observed to be larger than the MAE value in the prediction task, as shown in **Table 3**, indicating that some extreme outliers occur in the data. Again, the GNN achieved superior results in this case.

Through evaluation on collected datasets, we have verified the feasibility of the model under different data settings. After the model has been trained, it can be deployed on the relevant equipment to achieve real-time and online gait prediction and recognition.

TABLE 1 | Comparison of prediction results of different methods on the HuGaDB data.

| | | | Prediction task | | | | | |
| | | | One-wearer test | | | Generalization test | | |
Method	Learning rate	Epochs	MAE	MSE	Max error	MAE	MSE	Max error
GNN ($\alpha = 0.4$)	0.001	200	**0.0427**	**0.0144**	2.06	**0.091**	**0.0277**	**1.760**
LSTM	0.001	200	0.0428	0.01443	2.039	0.1001	0.035	2.126
CNN	0.001	200	0.055	0.0162	1.989	0.1424	0.0474	1.978
BP	0.001	200	0.0522	0.018	2.156	0.1159	0.0406	2.150

Bold values represent the best performance.

TABLE 2 | Comparison of recognition results of different methods on the HuGaDB data.

Methods	One-wearer test accuracy (%)	Generalization test accuracy (%)
RECOGNITION TASK		
GNN ($\alpha = 0.4$)	**100**	98.04
LSTM	95.67	92.78
BP	97.5	78.49
CNN	96.39	79.24
LightGBM	99.76	**99.33**
SVM	**100**	98.62

Bold values represent the best performance.

FIGURE 8 | Gait prediction by the GNN based on one-wearer gait data.
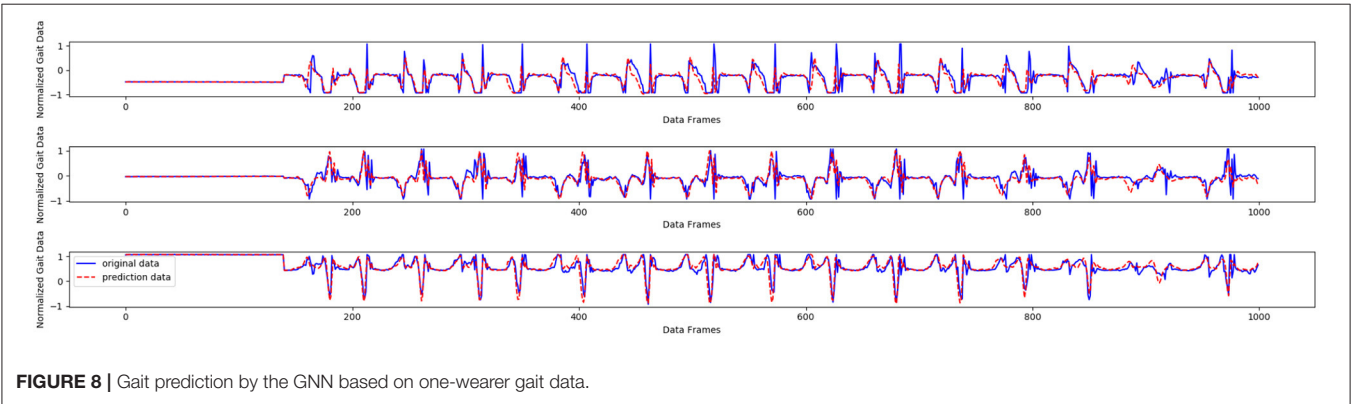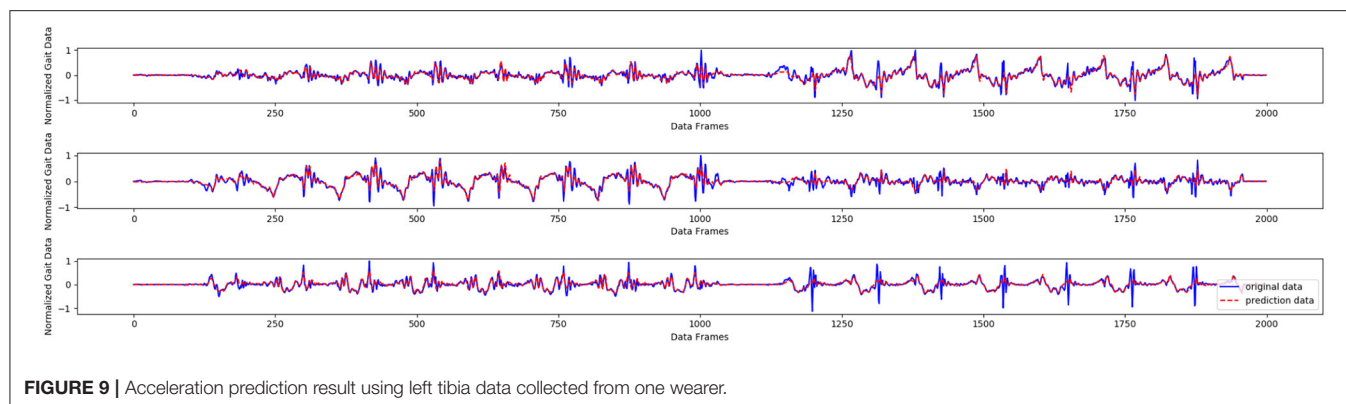
FIGURE 9 | Acceleration prediction result using left tibia data collected from one wearer.

TABLE 3 | Comparison of the prediction results of different methods using the collected data.

Methods	Learning rate	Epochs	MAE	MSE	Max error
PREDICTION TASK					
GNN	0.001	200	**0.1269**	**0.1314**	**7.28**
LSTM	0.001	200	0.1647	0.2885	9.56
CNN	0.001	200	0.3308	1.4429	15.28
BP	0.001	200	0.1290	0.1391	7.77

Bold values represent the best performance.

TABLE 4 | Comparison of the recognition results of different methods using the collected data.

Methods	Accuracy (%)
RECOGNITION TASK	
GNN	**98.81**
LightGBM	98.34
SVM	97.62
BP	91.68
LSTM	88.78
CNN	85.38

Bold values represent the best performance.

4. CONCLUSIONS

This article has proposed the GNN as a model for human-exoskeleton interaction. Comparisons of the GNN and other methods on the HuGaDB dataset show that the GNN consistently achieves superior performance. The results further demonstrate that the GNN's generalization performance is better than that of the other methods, despite the increase in the MAE and MSE. Because of the size of the dataset, only three wearers'

gait data were used to test the generalization ability. Including more gait data from different groups to train the network should enable even better prediction results to be obtained. For further evaluation of the method, gait data on complex movements were collected using an inertial-based motion capture device. By evaluating the GNN on the collected data, we find that it achieves efficient human gait prediction performance even without strong periodicity. Generally the GNN takes <2 ms to complete the task of gait recognition and prediction. Based on these results, it can be concluded that the GNN model can effectively recognize and predict human motion states.

DATA AVAILABILITY STATEMENT

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation, to any qualified researcher.

AUTHOR CONTRIBUTIONS

BF designed the network and wrote the manuscript. QZho and JS wrote the code and analyzed the results. FS, MW, and CX helped to improve the manuscript. QZha provided the real experimental data. All authors contributed to the article and approved the submitted version.

FUNDING

REFERENCES

Ackermann, M., and van den Bogert, A. J. (2010). Optimality principles for model-based prediction of human gait. *J. Biomech.* 43, 1055–1060. doi: 10.1016/j.jbiomech.2009.12.012

Aertbeliën, E., and Schutter, J. D. (2014). "Learning a predictive model of human gait for the control of a lower-limb exoskeleton," in *5th IEEE RAS/EMBS*

International Conference on Biomedical Robotics and Biomechatronics (BIOROB 2014) (São Paulo). doi: 10.1109/BIOROB.2014.6913830

Bai, S., Kolter, J. Z., and Koltun, V. (2018). An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *CoRR* abs/1803.01271.

Casale, P., Pujol, O., and Radeva, P. (2011). "Human activity recognition from accelerometer data using a wearable device," in *Iberian Conference on Pattern*

Recognition and Image Analysis (Barcelona). doi: 10.1007/978-3-642-2125 7-4_36

Chalvatzaki, G., Koutras, P., Hadfield, J., Papageorgiou, X. S., Tzafestas, C. S., and Maragos, P. (2018). On-line human gait stability prediction using lstms for the fusion of deep-based pose estimation and lrf-based augmented gait state estimation in an intelligent robotic rollator. *CoRR* abs/1812.00252. doi: 10.1109/ICRA.2019.8793899

Chereshnev, R., and Kertesz-Farkas, A. (2017). HuGaDB: human gait database for activity recognition from wearable inertial sensor networks. *arXiv* 1705.08506. doi: 10.1007/978-3-319-73013-4_12

Du, Y., Fu, Y., and Wang, L. (2015). "Skeleton based action recognition with convolutional neural network," in *2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR)* (Kuala Lumpur). doi: 10.1109/ACPR.2015.7486569

Ferrari, A., Bergamini, L., Guerzoni, G., Calderara, S., Bicocchi, N., Vitetta, G., et al. (2019). Gait-based diplegia classification using lsmt networks. *J. Healthc. Eng.* 2019, 1–8. doi: 10.1155/2019/3796898

Gross, R., and Shi, J. (2001). *The CMU Motion of Body (MoBo) Database*, Vol. 27. Technical Report.

Iwama, H., Okumura, M., Makihara, Y., and Yagi, Y. (2012). The OU-ISIR gait database comprising the large population dataset and performance evaluation of gait recognition. *IEEE Trans. Inform. Forens. Sec.* 7, 1511–1521. doi: 10.1109/TIFS.2012.2204253

Lea, C., Flynn, M. D., Vidal, R., Reiter, A., and Hager, G. D. (2016a). Temporal convolutional networks for action segmentation and detection. *CoRR* abs/1611.05267. doi: 10.1109/CVPR.2017.113

Lea, C., Vidal, R., Reiter, A., and Hager, G. D. (2016b). "Temporal convolutional networks: a unified approach to action segmentation," in *European Conference on Computer Vision* (Cham). doi: 10.1007/978-3-319-49409-8_7

Lin, L., Xu, B., Wu, W., Richardson, T., and Bernal, E. A. (2019). Medical time series classification with hierarchical attention-based temporal convolutional networks: a case study of myotonic dystrophy diagnosis. *arXiv* 1903.11748.

Martinez, J., Black, M. J., and Romero, J. (2017). "On human motion prediction using recurrent neural networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (Honolulu, HI). doi: 10.1109/CVPR.2017.497

MatthewDavies, E., and Bock, S. (2019). "Temporal convolutional networks for musical audio beat tracking," in *2019 27th European Signal Processing Conference (EUSIPCO)* (A Coruna), 1–5. doi: 10.23919/EUSIPCO.2019.8902578

Romero-Hernandez, P., de Lope Asiain, J., and Graña, M. (2019). "Deep learning prediction of gait based on inertial measurements," in *Understanding the Brain Function and Emotions. IWINAC 2019. Lecture Notes in Computer Science*, Vol. 11486, eds J. Ferrández Vicente, J. Álvarez-Sánchez, F. de la Paz López, J. Toledo Moreo, and H. Adeli (Cham: Springer), 284–290. doi: 10.1007/978-3-030-1959 1-5_29

Shotton, J., Fitzgibbon, A., Cook, M., Sharp, T., Finocchio, M., Moore, R., et al. (2011). Real-time human pose recognition in parts from single depth images. *Commun. ACM* 56, 1297–1304. doi: 10.1109/CVPR.2011.59 95316

Wang, J., Chen, Y., Hao, S., Peng, X., and Hu, L. (2018). Deep learning for sensor-based activity recognition: a survey. *Pattern Recogn. Lett.* 119, 1–9. doi: 10.1016/j.patrec.2018.02.010

Zhang, S., Wang, C., Wu, X., Liao, Y., and Wu, C. (2015). "Real time gait planning for a mobile medical exoskeleton with crutche," in *2015 IEEE International Conference on Robotics and Biomimetics (ROBIO)* (Zhuhai). doi: 10.1109/ROBIO.2015. 7419117

Zheng, S., Zhang, J., Huang, K., He, R., and Tan, T. (2011). "Robust view transformation model for gait recognition," in *18th IEEE International Conference on Image Processing, ICIP 2011* (Brussels). doi: 10.1109/ICIP.2011.6115889

Zhu, W., Lan, C., Xing, J., Zeng, W., Li, Y., Shen, L., et al. (2016). "Co-occurrence feature learning for skeleton based action recognition using regularized deep LSTM networks," in *AAAI'16: Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence* (Phoenix, AZ).

A Hybrid BCI Based on SSVEP and EOG for Robotic Arm Control

Yuanlu Zhu [1,2], Ying Li [1,2], Jinling Lu [1,2]* and Pengcheng Li [1,2,3]

[1] Wuhan National Laboratory for Optoelectronics, Britton Chance Center of Biomedical Photonics, Huazhong University of Science and Technology, Wuhan, China, [2] MoE Key Laboratory for Biomedical Photonics, School of Engineering Sciences, Huazhong University of Science and Technology, Wuhan, China, [3] Huazhong University of Science and Technology-Suzhou Institute for Brainsmatics, Suzhou, China

Brain-computer interface (BCI) for robotic arm control has been studied to improve the life quality of people with severe motor disabilities. There are still challenges for robotic arm control in accomplishing a complex task with a series of actions. An efficient switch and a timely cancel command are helpful in the application of robotic arm. Based on the above, we proposed an asynchronous hybrid BCI in this study. The basic control of a robotic arm with six degrees of freedom was a steady-state visual evoked potential (SSVEP) based BCI with fifteen target classes. We designed an EOG-based switch which used a triple blink to either activate or deactivate the flash of SSVEP-based BCI. Stopping flash in the idle state can help to reduce visual fatigue and false activation rate (FAR). Additionally, users were allowed to cancel the current command simply by a wink in the feedback phase to avoid executing the incorrect command. Fifteen subjects participated and completed the experiments. The cue-based experiment obtained an average accuracy of 92.09%, and the information transfer rates (ITR) resulted in 35.98 bits/min. The mean FAR of the switch was 0.01/min. Furthermore, all subjects succeeded in asynchronously operating the robotic arm to grasp, lift, and move a target object from the initial position to a specific location. The results indicated the feasibility of the combination of EOG and SSVEP signals and the flexibility of EOG signal in BCI to complete a complicated task of robotic arm control.

Keywords: hybrid brain-computer interface (BCI), electrooculography (EOG), robotic arm control, steady-state visual evoked potential (SSVEP), information transfer rates (ITR)

INTRODUCTION

Brain-computer interfaces (BCIs) are designed as a bridge to construct direct communication between the brain and external devices without relying on normal peripheral nerves and muscle tissue (Wolpaw et al., 2000). BCIs aim to provide people with severe motor disabilities an alternative to communicate and control external devices. Robotic arm control is one of the popular applications of BCI. Many studies have attempted to realize BCI for robotic arm control to improve the life quality of people with motor impairment (Pfurtscheller et al., 2010b; Gao et al., 2017; Khan and Hong, 2017). Considering the practical use of people with motor disabilities, the system design should focus more on the accuracy of command execution and the convenience of operation.

Electroencephalography (EEG) is one of the most widely used non-invasive BCI for its low cost, portability and high temporal resolution. Several types of physiological activation are usually chosen to generate the output commands of the EEG-based BCI, such as motor imagery (MI) (Wolpaw et al., 1991), P300 (Farwell and Donchin, 1988), and steady-state visual evoked potential

(SSVEP) (Cheng et al., 2002). Single modality which uses only one type of input signal usually has its own limitation in the number of commands and the classification accuracy. To promote the application of BCIs, several researches employed multiple modalities to improve the performance of the system by combing the advantages of different modalities, which is named as hybrid BCI (Pfurtscheller et al., 2010a).

As for the convenience of the system operation, electrooculography (EOG) is a good choice for its easy to execute and detect. EOG is the depolarization and hyperpolarization between retina and cornea caused by different eye movements, forming a potential difference between retina and cornea whose amplitude is larger than that of EEG and background physiological signals. Therefore, EOG can be easily and accurately detected using a few of electrodes around eyes. Compared to the conventional hybrid BCI most of which utilized the multiple types of EEG signals, the combination of EEG and EOG signals to construct a hybrid BCI can reduce the workload of users and makes the operation more convenient. The eye movements often used are blinking, winking, frowning, and gazing. Several studies used EOG in BCI to reflect the intention of subjects and to transmit commands to external devices. Nakanishi and Mitsukura proposed a wheelchair control system by using the voluntary eye blink (Nakanishi and Mitsukura, 2013). Ma et al. introduced a multithreshold EOG detection method and combined the EOG and P300 for robot control which used different eye movements to obtain the control commands and turn the stimulus on and off to enhance the performance of the system (Ma et al., 2015). He et al. proposed a hybrid BCI based on MI and EOG signals to operate a web browser (He et al., 2017). Wang et al. combined MI, P300, and EOG signals to asynchronously control a wheelchair (Wang et al., 2014). Huang et al. used EOG for button selection, MI for directional control, and combined computer vision for the control of an integrated wheelchair robotic arm system (Huang et al., 2019). Tan et al. applied autoencoder-based transfer learning in hybrid BCI for rehabilitation robot which composed of MI-based rehabilitation action, SSVEP-based menu selection, and EOG-based operation confirmation of cancellation (Tan et al., 2019).

To achieve asynchronous SSVEP-based BCI system, several studies distinguished the control state from idle state by using threshold criteria during the stimulus flashing (Ortner et al., 2011; Pan et al., 2013; Zhou et al., 2020). When the stimuli kept flashing since the start of experiments including control and idle states, asynchronous BCI used threshold criteria were susceptible to be incorrect activated due to the implicit attention to the flicker stimuli. Considering the effect of stimulus flicker on asynchronous detection, other studies applied a switch to activate or deactivate the stimulus flicker (Pfurtscheller et al., 2010b; Gao et al., 2017; Li et al., 2018). Pfurtscheller et al. used sequential MI-based brain switch to turn on or off the SSVEP-based BCI (Pfurtscheller et al., 2010b). The low classification performance of MI lead to a FAR with 1.46 per minute. Given the high signal-to-noise ratio (SNR) of EOG signals, Li et al. applied a single blink synchronized with a random flashing button as the switch of wheel chair control (Li et al., 2018). They used two consecutive

intended blinks as a start command with no false option occurred in static state and an intended blink as a stop command with a FAR of 0.18 per minute in the motion state. Due to the switch detection based on the synchronization of the flicker button and a single blink, the button was required to flash in the idle state.

In present study, to further decrease the FAR of asynchronous SSVEP-based BCI, we designed an EOG-based switch with no need for stimulus in idle state and combined the switch with a timely cancel command to effectively control a robotic arm. Through using the EOG-based switch to activate and deactivate the flicker stimuli, there was no need for extern stimulus in idle state which decreased visual fatigue caused by flashing and was more in line with perception of idle state. The FAR of the proposed asynchronous SSVEP-based BCI related only to the detection accuracy of EOG signals. Due to the high SNR of EOG signals, the detection of EOG is more accurately. We selected the triple blink as the EOG-based switch due to its ease of completion and low probability of occurrence in normal physiological situation. Moreover, we designed a cancel command based on a wink to make the subject be able to cancel the execution of a current command as needed. The control commands of the robotic arm were obtained by a SSVEP-based BCI, considering that SSVEP has gained a lot of attention in BCI for the reason of less training, high classification accuracy and information transfer rates (ITR). Fifteen buttons consist of the graphical user interface (GUI) of the SSVEP-based BCI, subjects were asked to focus on one of the fifteen buttons in a flash cycle to transmit corresponding control commands to the robotic arm. Given the relatively large number of stimuli, the default setting is to execute the feedback command, which was only canceled when the feedback stage recognized a wink from subject to helped control the robotic arm to complete the action more effectively. The algorithm of detecting SSVEP used was filter bank canonical correlation analysis (FBCCA) method (Chen et al., 2015a) which made use of the information in harmonic frequencies to improve classification accuracy. A multi-threshold method (Ma et al., 2015) was adopted to detect different eye movement waveforms. The experimental results showed the feasibility of the proposed system and the ability to complete a complicated task with a series of actions through the combination of EOG and SSVEP signals for robotic arm control.

MATERIALS AND METHODS

Subjects

Fifteen healthy volunteers (8 female; age 24.9 ± 2.5 years) with normal or corrected-to-normal vision participated in the experiments. Eleven subjects conducted two online experiments to evaluate the performance of the proposed hybrid EOG-SSVEP-based BCI. Six subjects conducted comparable experiments to evaluate the effectiveness of using EOG for command cancellation. All subjects are undergraduate and graduate students, and three of them have some experience with MI-based BCI experiments, while others are naive to BCI experiments. Before the experiments, each subject read and signed an informed consent form approved by the Human Subjects Institutional Review Board of Huazhong University of Science

and Technology. Subjects obtained a small compensation for participating in the experiments.

Data Acquisition System

In this study, EEG and EOG signals were recorded at sampling rate of 250 Hz with high-pass and low-pass filters of 0.1 and 250 Hz using a multichannel EEG system from Brain Products (BrainAmp, Germany). A total of nine electrodes, HEOR, Fp1, Pz, PO3, POz, PO4, O1, Oz, and O2 were placed according to the International 10–20 system (see **Figure 1**). Electrodes Pz, PO3, POz, PO4, O1, Oz, and O2 were used to collect SSVEP-based EEG raw signal, and the electrodes HEOR and Fp1 were selected to record the EOG signal. The electrode on the forehead (AFz) was used as ground and the reference electrode was positioned on the vertex (Cz). All electrodes impedances were maintained below 10 KΩ.

The visual stimuli were presented on a 23.8-inch LCD screen with a resolution of 1,650 × 1,080 pixels. The refresh rate of the screen was 60 Hz. All subjects were arranged to seat in a comfortable chair in front of the visual stimulus computer at a distance of ∼70 cm in a quiet room. The visual angle between the robotic arm and the monitor was 45°. This arrangement allowed subjects to look at both the monitor and the movement of the robotic arm.

GUI

The GUI was designed to ensure the effective and accurate control and operation of the robotic arm. As illustrated in **Figure 2**, the GUI was composed of two sessions: the switch interface which displayed "Please blink three times rapidly to open/close the SSVEP-based interface" in the center of the screen to prompt the subjects to use the EOG-based switch, and the SSVEP-based interface consisted of a 3 × 5 flashing stimulus matrix representing 15 commands which were designed to control the robotic arm for the grasp and move actions. Visual flashing buttons of the SSVEP-based interface were presented using a sampled sinusoidal stimulation method (Manyakov et al., 2013; Chen et al., 2014). The size of each button was 150 × 150 pixels. All buttons flashed between green and blue under black background to reduce visual fatigue (Takano et al., 2009; Chen et al., 2017; Floriano et al., 2018). The horizontal and vertical distance of each adjacent buttons was 150 pixels. The range of the stimulus frequency for the fifteen visual flashing buttons in the proposed study was chosen from 8 to 15 Hz with an interval of 0.5 Hz because of its relatively high response in their corresponding SSVEP signal (Chen et al., 2018). The stimulus paradigm of the BCI was realized by using the Psychophysics Toolbox Version 3 (Brainard, 1997) on MATLAB (MathWork, Inc).

As shown in **Figure 2B,** there were two lines of text at the top of the SSVEP-based interface to assist the subjects in getting the real-time status of the robotic arm. The first line of text indicated the current programmed moving step of the robotic arm, and the second line of text displayed the corresponding configured position of each axis of the robotic arm. The left four columns of buttons corresponded to different directions of the movements of the robotic arm. And the robotic arm had a total of six axis (S1,

S2, S3, S4, S5, and S6). For a specific direction, "S1" represented the rotation of the robotic arm in the x-y plane, and "S2," "S3," and "S4" allowed the robotic arm to move to different degrees along the z-axis. "S5" was used to rotate the claws. "S6" drove the robotic arm to clamp or loosen. Among them, "S1+" and "S1−" indicated the opposite direction, respectively, and others were the same. In order to make an effective operation, two buttons ("step+" and "step−") were added to change the moving step of the robotic arm movements in different direction. "R" was utilized to return the robotic arm to its original position. **Figure 2C** showed the stimulation frequency of each target.

System Configuration Description

For practical use, the design of multitask makes the control and operation of the BCI more flexible and versatile. This study combined the eye movements and SSVEP to realize an asynchronous hybrid BCI. As illustrated in **Figure 3**, the proposed asynchronous EOG-SSVEP-based robotic arm control system mainly consisted of four hardware components: an EEG acquisition device, a visual stimulus computer, a robotic arm, and a host computer used as data online processor. The EEG signals were recorded and transmitted to the host computer with synchronous event triggers sent from the visual stimulation computer for real-time preprocessing and classification. The visual stimulation computer was not only utilized to present the stimulus paradigms and online visual feedback but also for translating relevant commands to the robotic arm via serial communication protocol. Six axis (ZX-361S) and an open source STM32 control board composed of the robotic arm which was able to be directly and easily controlled by the SSVEP-based interface through serial port. The manipulating angle of each axis was configured in the range of 0° to 180°, and the rotation speed of each axis could be adjusted according to the moving step which was set by the subjects.

The system flowchart can be seen in **Figure 4**. After the start of the experiment, EEG data recorded from the subject are first preprocessed to remove the baseline drift and the influence of the environment. And then the SSVEP interface can be activated only when a triple blink from the subject is detected by the system. Otherwise the system will maintain in the switch interface. To effectively detect EOG signal when subjects blink three times rapidly, a calibration process was conducted before online experiment to determine the appropriate online threshold for each subject. When the SSVEP-based interface is activated, the SSVEP signal and the triple blink are detected in parallel. Subjects are allowed to blink three times rapidly when they hope to switch off the flash of the buttons and return to the EOG-based switch interface. If no triple blink is detected, the classification of the SSVEP signal will be transmitted to the visual stimulus computer as a feedback to the subject. And a robotic control command corresponds to the specific classification result. Once the subject wants to cancel the command sending to the robotic arm, he or she is asked to execute a wink after the occurrence of the feedback. If no wink is detected at the feedback phase, the robotic arm will execute the relevant command and then another flashing cycle begin to generate another new command.

FIGURE 1 | Location of nine electrodes for EEG recording.
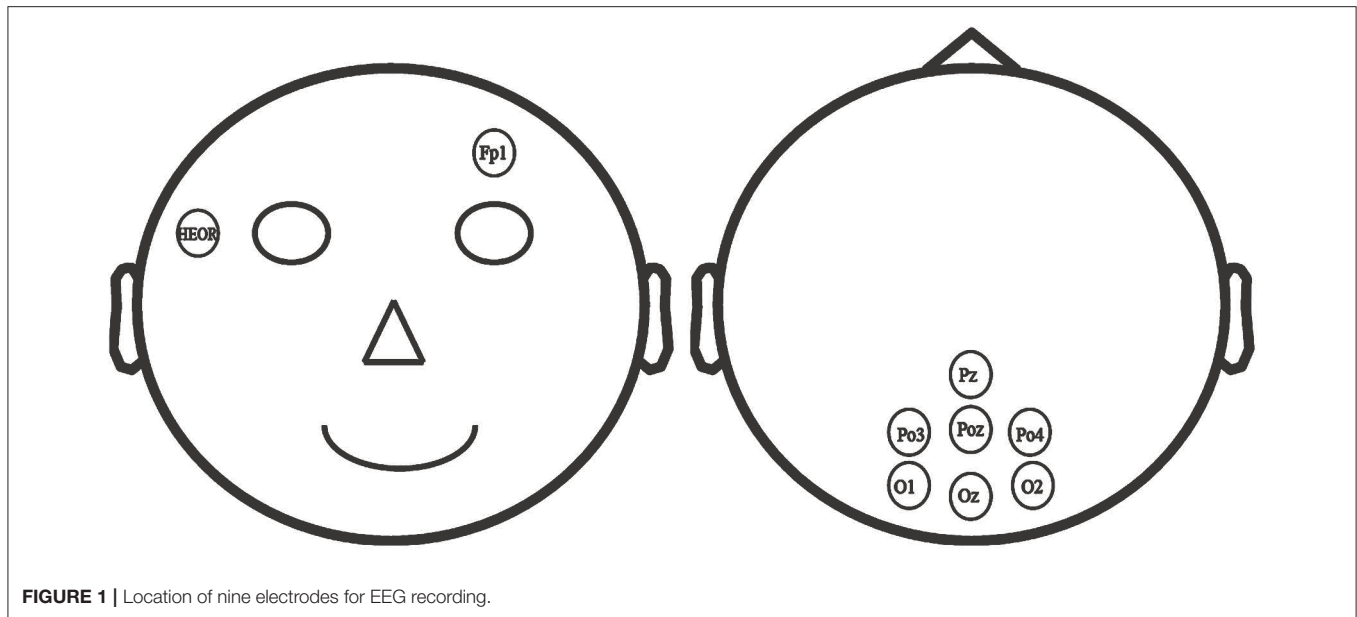
Data Processing and Detection Algorithm

In this study, the EEG and EOG data were processed and fed back in real time. The recording data were firstly preprocessed to minimize the impact of external environment and motion artifacts. The 50 Hz interference of power supply was eliminated by a notch filter. The EEG signals were then re-referenced to the arithmetic average across all recording channels. The detection algorithm was mainly composed of two parallel parts: EOG data analysis and SSVEP data analysis.

EOG Data Analysis

The proposed asynchronous hybrid BCI allowed subjects to decide when to start and stop the control of the system by an EOG-based switch which was designed by the detection of a triple blink from the Fp1 channel. The detection is conducted every 100 ms from the beginning of the experiment. For real-time analysis, we use a sliding window with a length of 1,200 ms with an interval of 100 ms. Furthermore, a wink was used to cancel the command in feedback phase, and the detection for wink is based on a segment data with a length of 2000 ms which contained the feedback and remind phase from the HEOR channel. The length of data is set in consideration of the reaction time of subjects. The detection method for the triple blink and wink is based on a multithreshold method described in Ma et al. (2015). For each detection, first, a segment data is extracted according to the window set for different eye movements. Then, on the purpose of removing physiological and environmental noise, the extracted data are bandpass filtered within the range of 0.1 to 15 Hz, after that the first-order difference operation is employed to get features of the eye movements as follows:

$$f'(n) = f(n) - f(n-1)$$

Where n is the sampled points, $f(n)$ is the relevant original value, and $f'(n)$ refers to the differential value of the original

data at point n. There are several features abstracted from the differentiated waveform for later analysis, e.g., the maximum peak value, the minimum peak value, the maximum amplitude, and the duration of the eye blink. **Figure 5** showed the raw and differential EOG data from channel HEOR in a trial that prompted the subject to wink during the calibration session. The differential EOG data of a wink contained a positive and negative wave, and its amplitude was much larger than the fluctuation of the EOG when no wink is performed. In order to extract the signal from a wink, we need to set a minimum value (Vp) for a positive wave. The signal is considered to be a positive wave when its voltage is greater than the Vp. Likewise, we need to set a maximum value (Vn) for the negative wave. The signal is considered to be a negative wave when its voltage is less than the Vn. When a positive wave followed by a negative wave is satisfied, to avoid recognizing the rest period signal as a wink, it is also necessary to satisfy that the amplitude of the original EOG (A) in the time period (D) from the beginning of the positive wave to the end of the negative wave is greater than the amplitude (Amp) of the initial setting, and the length of the time period (D) needs to be greater than the limited minimum duration (Dmin) and less than the limited maximum duration (Dmax) of the initial setting, then it is considered as a wink. The principle of a successful eye blink detection are as follows:

$$i = \begin{cases} 1, & \text{if } A \geq \text{Amp}, \ D_{min} \leq D \leq D_{max} \\ 0, & \text{otherwise} \end{cases}$$

Where i represents the detection result of a wink. If the features of the EOG waveform satisfy all condition, i is equal to 1 which means a successful wink detection, otherwise i is equal to 0 manifests no intentional wink was detected. Thus, the recognition of a wink requires the initial setting of five thresholds,
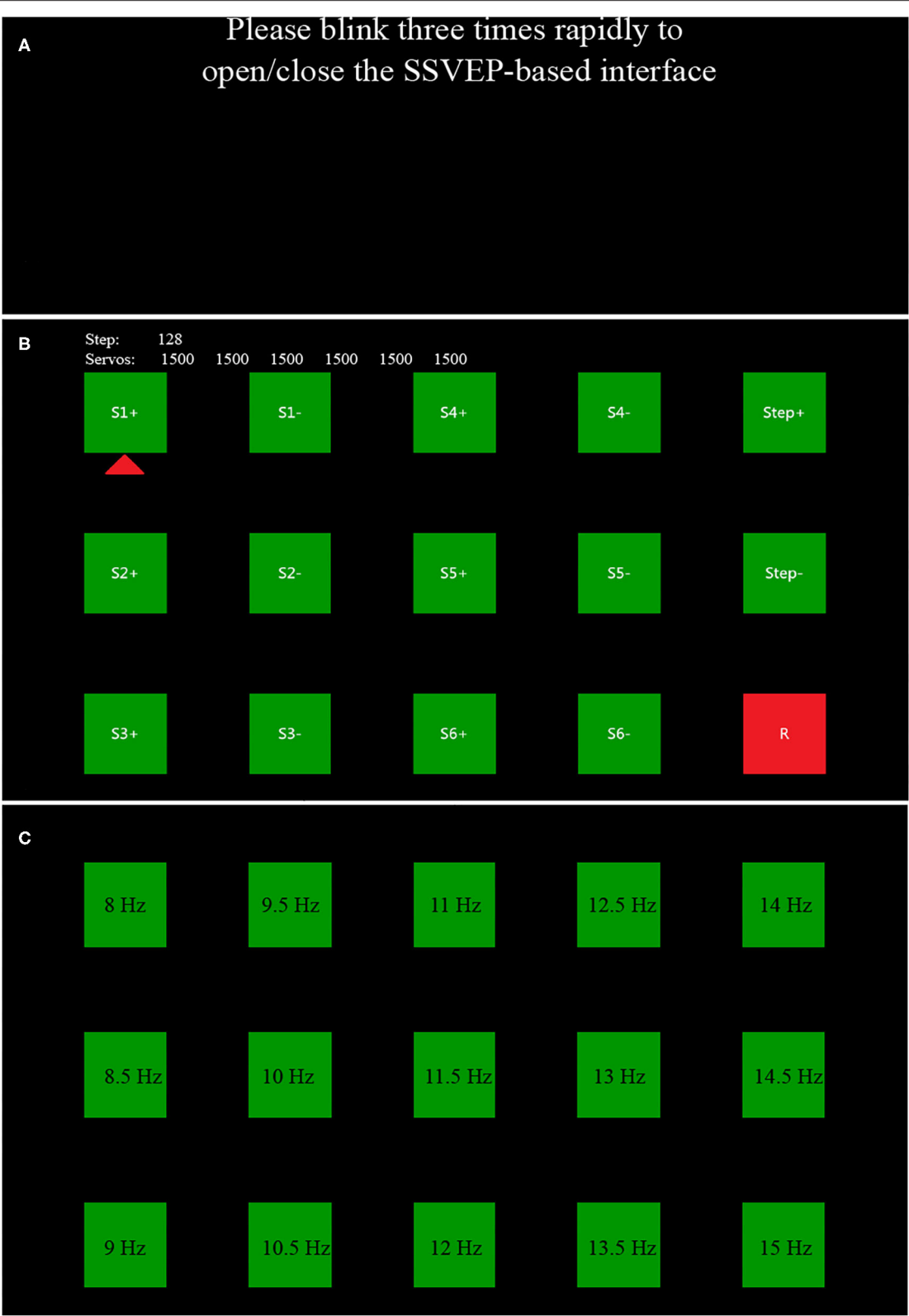
FIGURE 2 | The GUI of EOG-based switch **(A)** and SSVEP-based BCI **(B)**. In **(C)**, a 3 × 5 flashing stimulus matrixes labeled with different stimulus frequency represents a total of 15 commands for the robotic arm control.

FIGURE 3 | Schematic configuration of the proposed hybrid BCI for robotic arm control.
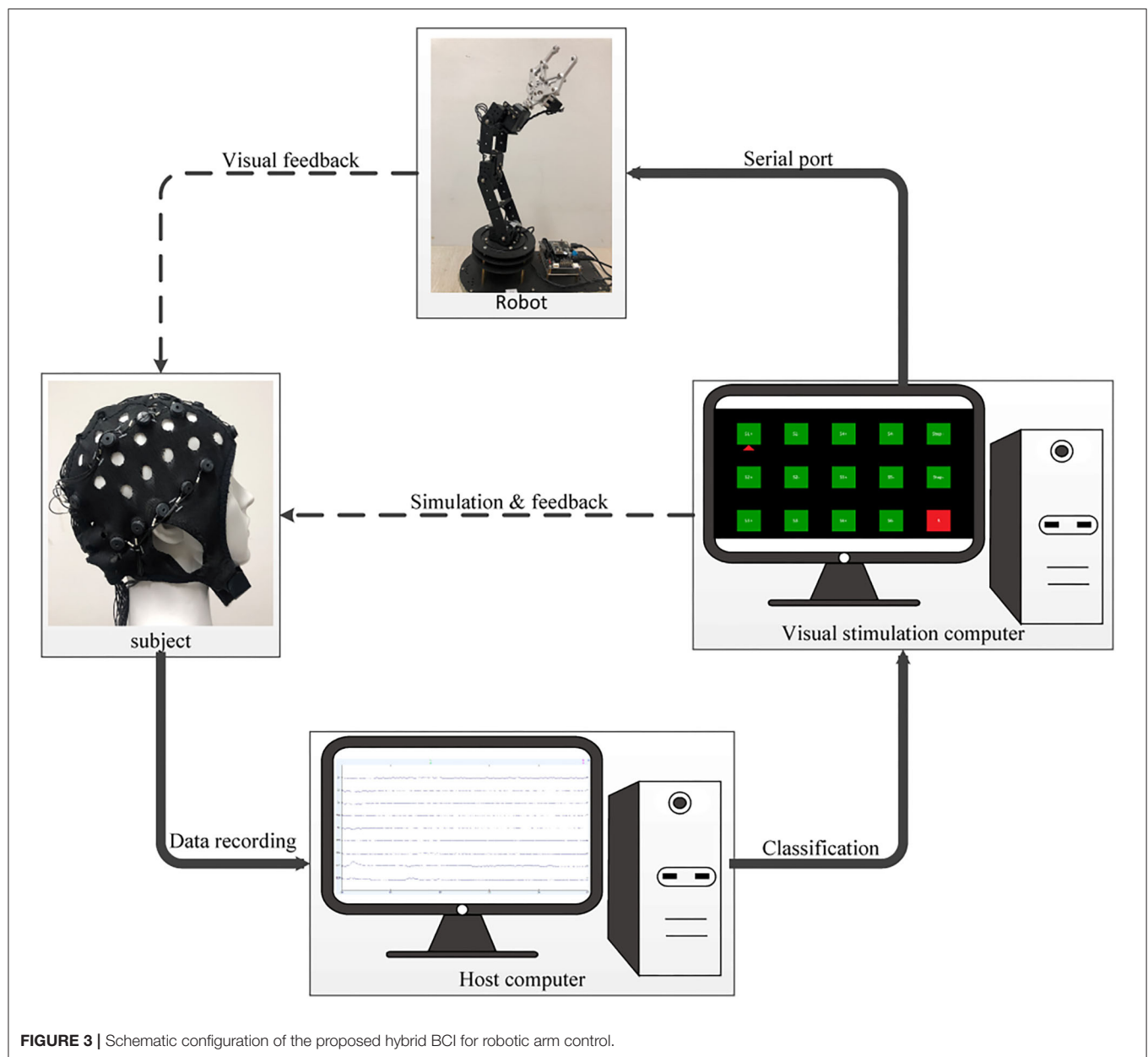
including Vp, Vn, Amp, Dmin, and Dmax. Since the duration of a wink does not vary much between subjects, Dmin and Dmax were set to 0.1 and 0.6 s for all subjects, respectively. The other three thresholds are influenced by the way each subject winks, so it is necessary to set specific thresholds for each subject in order to accurately identify the wink of each subject.

The basic algorithm for identifying triple blink is the same as the detection of a wink, except that it is considered as the triple blink only when three consecutive blinks are recognized within a limited time window length which is set to 1,200 ms. The **Figure 6** showed the raw and differential EOG data from channel Fp1 in a trial that prompted the subject to conduct triple blink during the calibration session. Therefore, to effective identify the intentional eye movements in online experiments,

a calibration process was asked to conduct for each subject first to obtain the thresholds required in the detection algorithm mentioned above.

SSVEP Data Analysis

In this study, the flash of the SSVEP-based interface was activated by the EOG-based switch. Once activated, the data epochs were extracted according to the event triggers for subsequent classification and the control of the robotic arm. Previous studies have shown that SSVEP induced by periodic visual stimulus contains brain response at the stimulus frequency and its harmonic and sub-harmonic frequencies (Herrmann, 2001). This study adopted the filter bank canonical correlation analysis (FBCCA) method (Chen et al., 2015a,b) which can effectively
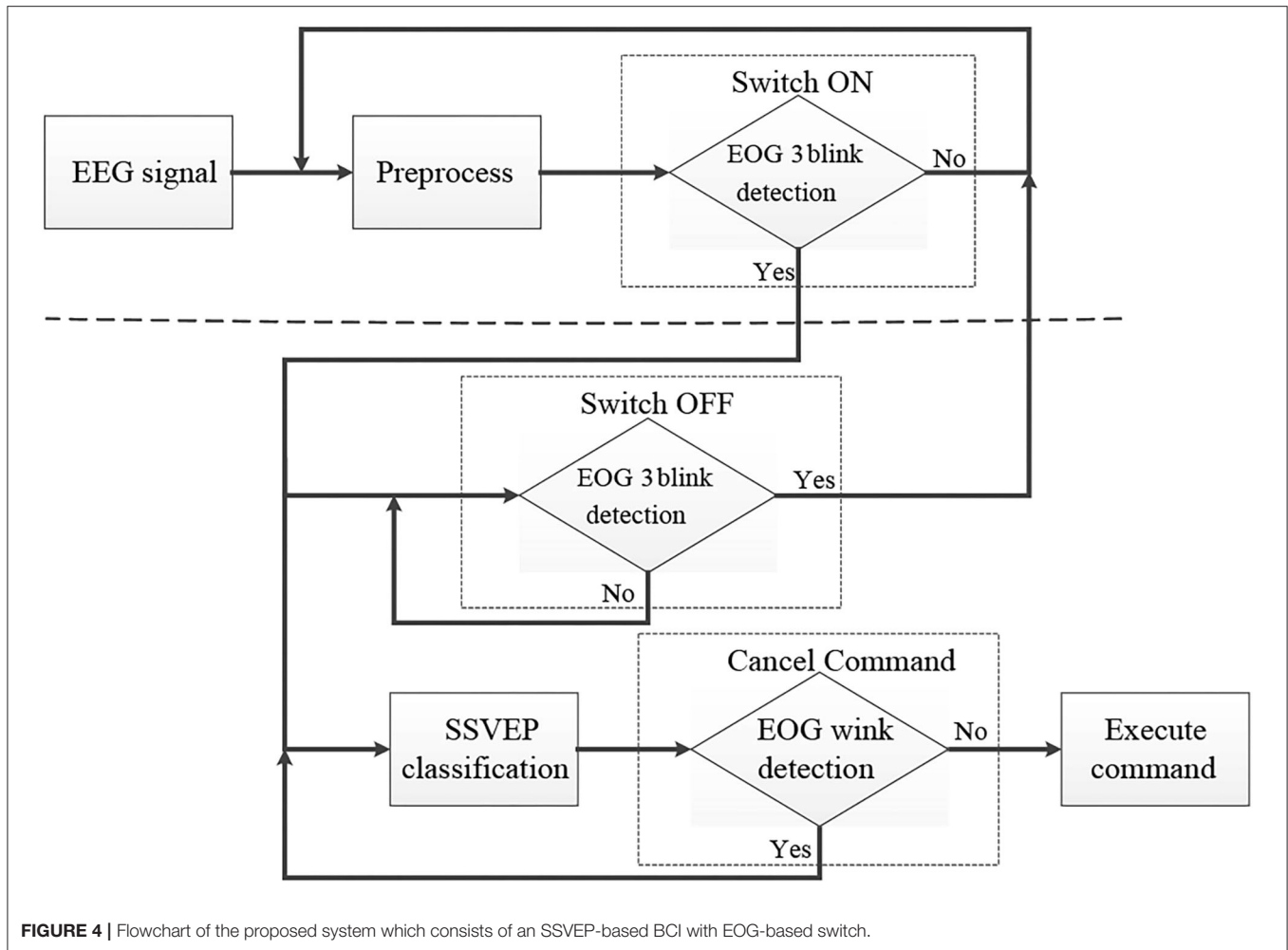
FIGURE 4 | Flowchart of the proposed system which consists of an SSVEP-based BCI with EOG-based switch.

utilize information in harmonic frequencies to enhance the detection of SSVEP. FBCCA method is mainly composed of three steps. First, a filter bank which consists of several bandpass filters decomposes the SSVEP data epochs into sub-band components. Second, an canonical correlation analysis (CCA) approach which has been widely adopted in BCI for SSVEP detection (Bin et al., 2009) is applied to get the correlation between sub-band components and predefined sinusoidal reference signals. Last, appropriate feature vectors are calculated for the target identification. In this study, we used CCA and FBCCA methods for the classification of SSVEP signals, and applied paired t-test for statistical analysis to evaluate the performance of CCA and FBCCA methods.

Classification accuracy and ITR were used to evaluate the performance of the proposed system. ITR was calculated according to the follow equation (Wolpaw et al., 2002):

$$ITR = \frac{60}{T}\left(log_2 N + Plog_2 P + (1\text{-P})\, log_2[\frac{1-P}{N-1}]\right)$$

Where T is the time it takes to output a command, including the time of the gaze shift, stimulus flicker, and feedback phase,

N is the total number of targets ($N = 15$), and P is the classification accuracy.

Calibration Process

For the purpose of effectively detecting the eye movements in real time, a calibration process is acquired to determine thresholds of different eye movements for each subject before online experiments. Appropriate detection thresholds allow the system to have short response time and high accuracy which make the system more reliable and flexible. During the calibration process, the paradigm of a trial consists of three parts. First, a fixation cross appears in the center of the visual stimulation screen for 1 s to prompt the subject to get ready for the task, and then the screen shows a text of "triple blink" or "wink" for 4 s to remind the subject to blink three times rapidly or wink in task period, after that 1 s black screen is displayed as a rest period. A total of 20 trials for triple blink and 20 trials for wink were designed for each subject to collect the datasets for the calculation of respective detection thresholds and to train subjects to be familiar with the eye movements simultaneously.

For offline processing, the recording data for triple blink and wink were extracted from all trials for thresholds calculation. We
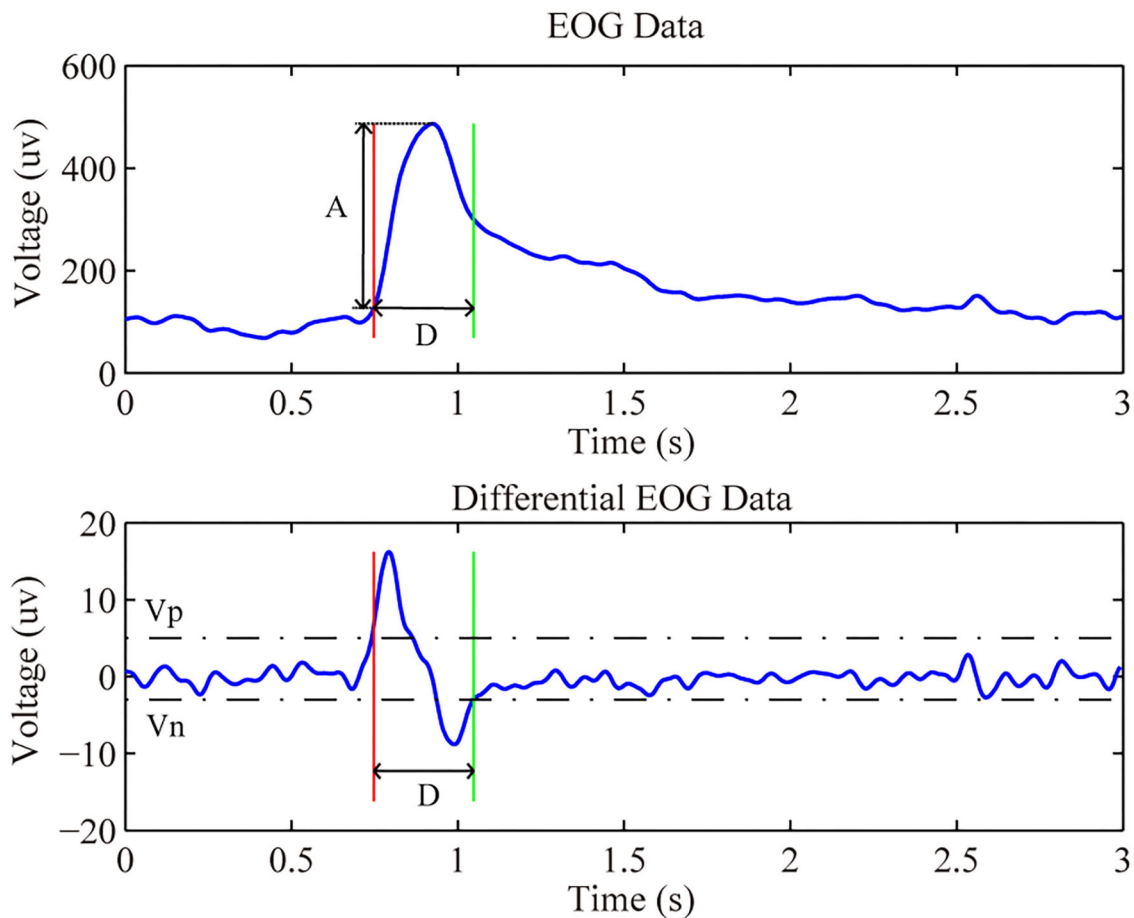
FIGURE 5 | The raw and differential EOG data of a wink.

used the detection algorithm described in the EOG data analysis to get the specific thresholds of triple blink and wink for each subject. Specifically, we calculated the first-order difference of the bandpass filtered data. In order to identify eye movements in task period, a predefined experiential threshold was used for the sampled data. The features of the eye movement waveforms were then computed. We removed the features of unqualified samples which only contained the motion artifacts but did not eye movements. The thresholds of online experiments was decided by the remaining features. The main thresholds to be obtained were Vp, Vn, and Amp.

Experiments

Before the online experiments, a calibration session mentioned above was carried out to determine appropriate online thresholds for each subject. Then two online experiments were performed using our asynchronous hybrid BCI system. One was a cue-based experiment, and the other was a self-paced operation of the robotic arm to conduct with a complicated task.

The Cue-Based Experiment

This experiment contained eight blocks which was designed to assess the performance of the proposed system and train subjects

to be familiar with the procedures of the system control. Each block consisted of two parts: the operation of the EOG-based switch and the gaze of the cue-based SSVEP interface for robotic arm control. Besides, in order to evaluate the capability of the use of winking to cancel the command in feedback phase, before the flicker of SSVEP-based interface, the clue given in GUI requested the subjects to wink or not to wink in the feedback phase of the current block. The clue indicating the request of a wink appeared every 2 blocks. A total of 4 blocks contained the data of winks. At the beginning of each block, the screen displayed the switch interface to prompt subjects to blink three times rapidly to activate the flash of the cue-based SSVEP interface. And then a cue with a red triangle appeared under one of the fifteen buttons with a pseudo-random order. There were a total of fifteen trials for the test of the SSVEP-based interface. Each trial with a duration of 5.5 s consisted of a remind phase for 0.6 s and a stimulation phase for 3 s and a feedback phase for 1.9 s. Subjects were asked to shift their gaze to the button indicated by the cue in the remind phase, and focus on the button in the stimulation phase to ignore the influence of other buttons as much as possible. In the feedback phase, a button was marked in red color according to the real-time classification. Subjects
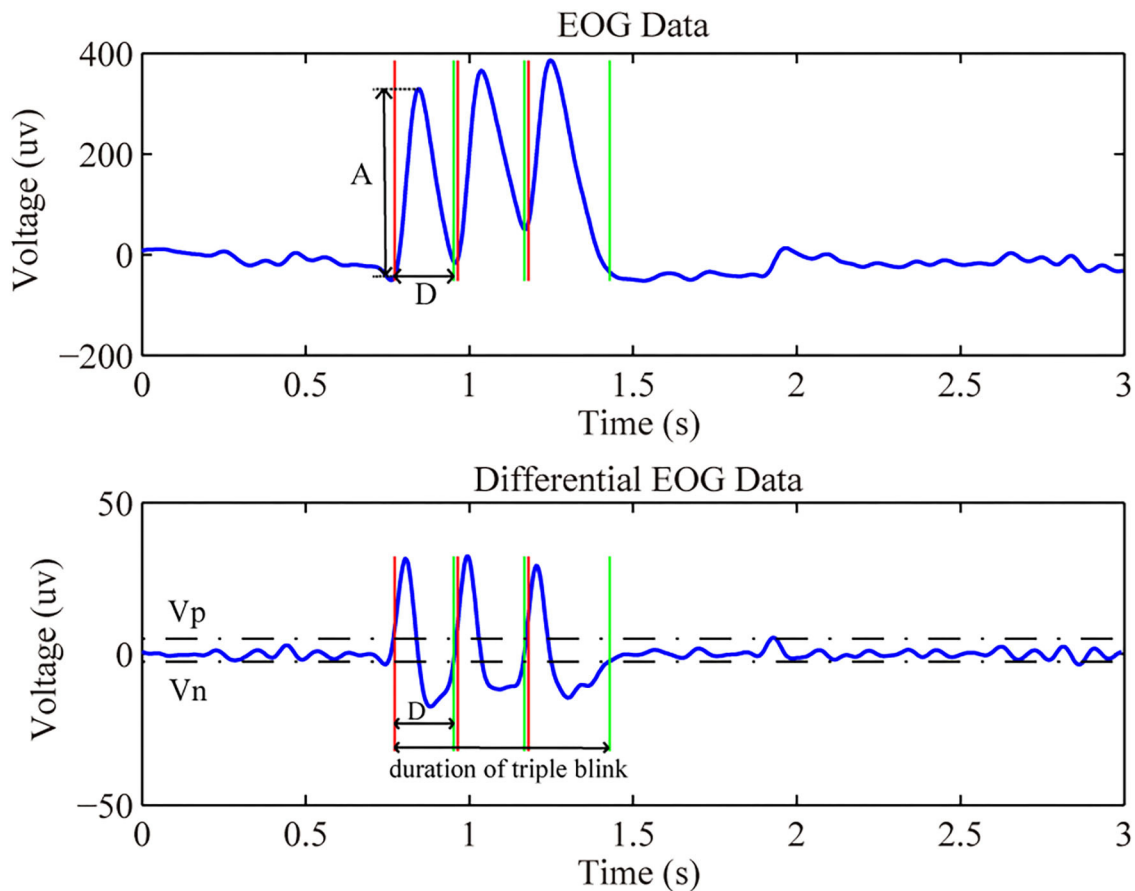
FIGURE 6 | The raw and differential EOG data of triple blink.

decided whether to use a wink to cancel the command based on the clue given in the start of the GUI. If there was no wink in the feedback phase, the corresponding command was transmitted to the robotic arm for motion control. When all buttons flickered once, the switch interface was appeared to remind subjects to use a triple blink to deactivate the flash of the SSVEP-based interface and take a break before the next block.

Asynchronous Robotic Arm Operation

In this experiment, the commands were selected by subjects to execute a series of sequential actions without visual cues. In order to maintain the difficulty of the experiment, the start and end location of the target object were consistent for each subject. Moreover, the robotic arm was reset to its initial position before the start of each operation. For the procedures of the experiment, in the first step, subjects used a triple blink to activate the flash of the SSVEP-based interface when they were ready to control the robotic arm. The paradigm of the SSVEP-based interface was mostly the same as the cue-based experiment except that there is no visual cues in the remind phase. Each trial started with the appearance of all stimuli in static state without visual cues, which lasted 0.6 s for attention and gaze shifting. Then the stimulus began to flicker with a duration of 3 s. Last, the screen

displayed the online feedback for 1.9 s, giving the subjects a chance to decide whether he or she will cancel the corresponding command or not. A wink after the appearance of the feedback could be identified as a canceled intention. Subjects were asked to operate the robotic arm to perform a series of actions of moving, grasping, lifting, and placing by gazing the SSVEP-based interface. After placing the target object to the specific location, the reset command was used to reset the robotic arm by gazing the "R" button twice before the next operation. The task execution time was recorded by the host computer based on the synchronous event triggers. To avoid misidentification, the reset command was executed only when the "R" button was detected in two consecutive trials. During the experiment, subjects were able to use a triple blink to deactivate the flash of SSVEP-based interface whenever they needed to rest and adjust.

RESULTS

As for the cue-based experiment, the classification accuracy and ITR were calculated to evaluate the performance of the SSVEP-based BCI. The false activation rate (FAR) which meant the rate of false triggering (Wang et al., 2014) was used to assess the

TABLE 1 | Results of EOG in cue-based experiment.

Subject	Triple blink FAR (event/min)	Wink TPR (%)	Wink FPR (%)
S1	0	95.56	0
S2	0	95	0
S3	0	95	1.67
S4	0	100	0
S5	0	96.67	0
S6	0.068	96.67	0
S7	0	96.67	1.67
S8	0.066	93.33	0
S9	0	95	1.67
S10	0	86.67	0
S11	0	78.33	0
Mean ± SD	0.01 ± 0.03	93.54 ± 6.00	0.46 ± 0.78

TABLE 2 | Results of SSVEP in cue-based experiment.

Subject	Accuracy (%)	ITR (bit/min)
S1	96.67	39.56
S2	100	42.62
S3	86.67	31.19
S4	93.33	36.57
S5	100	42.62
S6	85.83	30.32
S7	100	42.62
S8	83.78	28.91
S9	97.5	40.33
S10	88.33	33.44
S11	80.83	27.61
Mean ± SD	92.09 ± 7.20	35.98 ± 5.88

efficiency of the EOG-based switch. The FAR was calculated by dividing the number of false identifications of the triple blink during the stimulus flicker by the duration time of the cue-based experiment. The false positive rate (FPR) and true positive rate (TPR) of the wink were computed to evaluate the reliability of the cancellation of commands. There were four blocks in the cue-based experiment that prompted the subject for a wink to cancel the execution of the current command after feedback occurrence in each trial, and the other four blocks did not require a wink in the feedback phase. Thus, the TPR was calculated by dividing the number of winks identified in the four blocks that required a wink in the feedback phase by the total number of trials in those four blocks. The FPR was calculated by dividing the number of winks identified in the four blocks that did not require a wink during the feedback phase by the total number of trials in the four blocks. The results in **Table 1** showed that the EOG-based switch resulted in a very low FAR with average 0.01 event per minute for all subjects which meant that the switch had good stability and reliability. Subjects were able to use the switch by a triple blink to stop the flicker of buttons in idle state to reduce visual fatigue. The detection of a wink in the feedback phase resulted in an average TPR of 93.54% and FPR of 0.46%, which indicated the capability of using a wink to cancel the command. The efficient detection of the wink in the feedback stage made it more convenient and effective for the robotic arm control. As for SSVEP-based BCI, the results in **Table 2** showed that the proposed system worked well in robotic arm control and acquired an average accuracy of 92.09% and the average ITR was 35.98 bit/min. Therefore, the results of the cue-based experiment illustrated the potential of the hybrid BCI to perform complex tasks in practical applications.

After the cue-based experiment, subjects were familiar with the procedures of the proposed hybrid BCI, and then they were required to asynchronously utilize the proposed system to operate the robotic arm to perform a complicated task three times by grasping, lifting, and moving a target object (i.e., a little doll) from the initial position to a specific location. **Figure 7** showed the process of controlling the robotic arm to complete

the specified actions. In the experiment of self-paced operation of the robotic arm, all subjects succeeded in asynchronously grasping and moving the target object from the initial position to the specific location by directly controlling the robotic arm through the hybrid BCI system. To evaluate the efficiency of the hybrid BCI in performing the complex tasks, we recorded the completed time and total number of commands of each subject in the operations of the robotic arm. **Table 3** showed the results of the asynchronous experiment for operating the robotic arm through the hybrid EOG-SSVEP-based BCI. Since it was a complicated task for subjects to conduct, the numbers of commands and time required to complete the task were different for each subject, which were related to lots of factors, such as the classification accuracy, the focus and concentration on the task, the planning and grasp strategy, and the proficiency in the robotic arm operation.

DISCUSSION

This study attempted to realize an asynchronous hybrid BCI for robotic arm control through the combination of an EOG and SSVEP signals in BCI. Compared with synchronous BCI, the proposed system used EOG-based switch to deactivate the flash to rest or activate the flash to operate the robotic arm whenever they wanted which made the system more flexible and convenient. Additionally, EOG-based timely cancel command allowed users to control the robotic arm to complete a complicated task with a series of actions more effectively. Previous studies which proposed asynchronous SSVEP-based BCI mainly used a conventional threshold method to distinguish the control state from idle state (Cheng et al., 2002; Pfurtscheller et al., 2010b). In these studies, the buttons in the GUI continued to flash from the beginning of the experiment even when subjects were in idle state, which was easy to cause visual fatigue. Several researches designed novel methods to improve the performance of the asynchronous SSVEP-based BCI. Pan et al. proposed asynchronous SSVEP-based brain switches using a pseudo-key-based approach to improve the discrimination between control
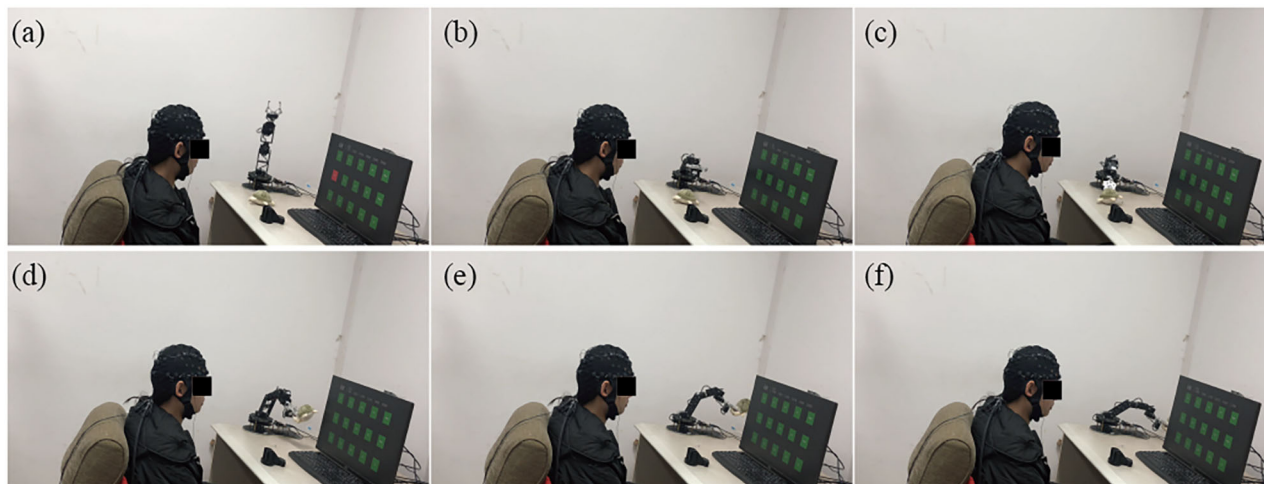
FIGURE 7 | The process of operating the robotic arm to grasp, lift, and move a target object from **(a–f)**.

and idle states (Pan et al., 2013). Pfurtscheller et al. used an MI-based brain switch to achieve self-paced operation of an SSVEP-based orthosis control system (Pfurtscheller et al., 2010b). Tomita et al. proposed of a bimodal BCI using simultaneously NIRS and EEG signals to estimate whether the subject is in idle or active mode (Tomita et al., 2014).

In this study, we chose EOG as the switch signals to either activate or deactivate the flash of the SSVEP-based interface for asynchronous operation of the robotic arm based on the intention of subjects. EOG-based switch in asynchronous SSVEP-based BCI did not need for stimulus in idle state when compared with asynchronous system used the threshold criteria. No stimulus in idle state help relieve fatigue. Moreover, compared with MI-based brain switch and the use of fNIRS signals, the EOG-based switch has the advantage of short response time and high SNR which makes it accurately distinguish the control state from idle state to decrease the FAR in the potential applications. But EOG-based switch also has its limitations. For the experiments lasting for a long time, the major challenge is that subjects may confuse the intended and unintended eye blinks when they get fatigue. Therefore, there is a need for future work to design a simpler and special switch mode, and the improvement of detection algorithm for different eye movements is also helpful. Furthermore, the present study used the SSVEP-based BCI to select specific actions performed by the robotic arm. SSVEP has the advantage of less training and relative high SNR, but the challenge for SSVEP is that it is easy to cause fatigue. In order to reduce user fatigue, we set the buttons to flash between green and blue under black background.

Additionally, the proposed system allowed subjects to timely cancel the command in feedback phase by a wink to effectively operate the robotic arm to complete a series of grasping, moving, and lifting actions. We implemented a comparable experiment to evaluate the effectiveness of using a wink to cancel command. Six subjects participated in the comparable

TABLE 3 | Results of asynchronous robotic arm operation.

Subject	Total number of commands	Completion time (s)
S1	71	394.26
S2	51	284.14
S3	83	473.53
S4	62	349.02
S5	55	305.37
S6	104	586.25
S7	36	197.54
S8	66	371.79
S9	58	324.27
S10	81	474.47
S11	89	500.04
Mean ± SD	68.73 ± 19.38	387.33 ± 112.43

experiment. Subject S9 and S11 also participated in the previous cue-based experiment. After conducting the calibration process to obtain the thresholds required for EOG detection, each subject was trained to be familiar with using the hybrid SSVEP-based BCI system to perform the grasping task described in the section "Asynchronous Robotic Arm Operation." And then each subject was asked to perform the grasping tasks with and without the capability of using a wink to cancel the feedback commands three times each. Considering that the execution sequence of tasks might influence the completion of the task, three subjects conducted the grasping tasks with a wink to cancel commands firstly, then without a wink to cancel commands, while the other subjects performed the grasping tasks in reverse sequence. The average number of commands executed by the robotic arm during the grasping task with and without a wink to cancel the commands for each subject was shown in **Figure 8**. The results indicated that using a wink to cancel the inappropriate commands in the grasping task significantly declined the number
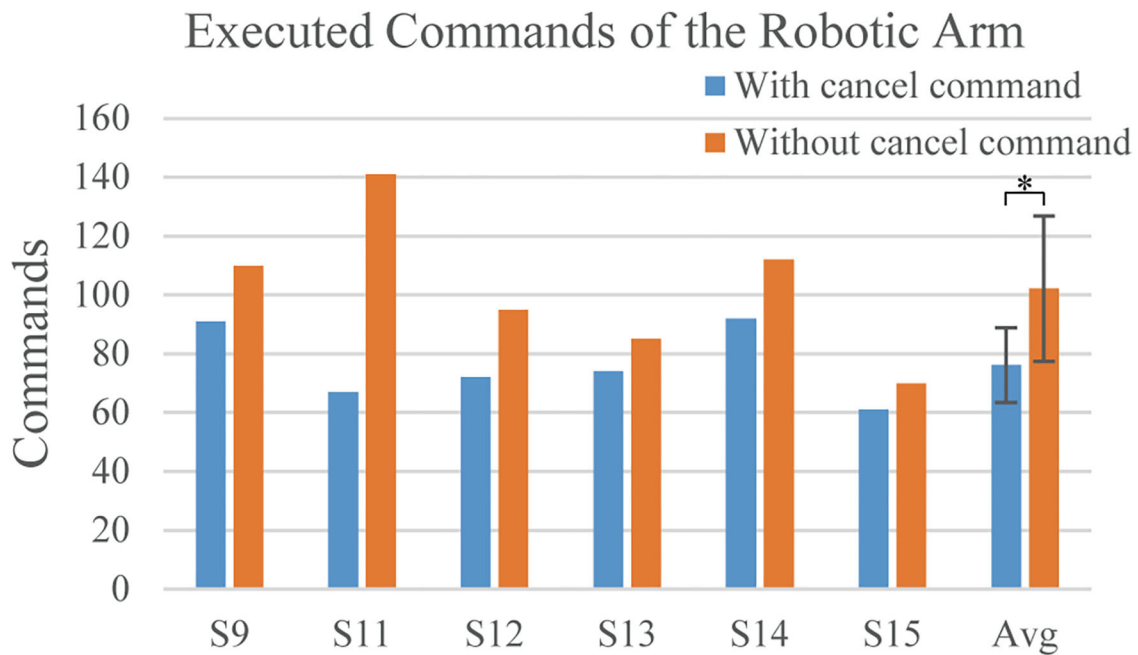
FIGURE 8 | The number of commands executed by the robotic arm for each subject in the grasping task with (blue bar) and without (orange bar) the use of a wink to cancel the feedback command. Avg indicated the average result of all subject. The error bar indicated the standard deviation. We used the paired t-test. *indicated the $p < 0.05$.
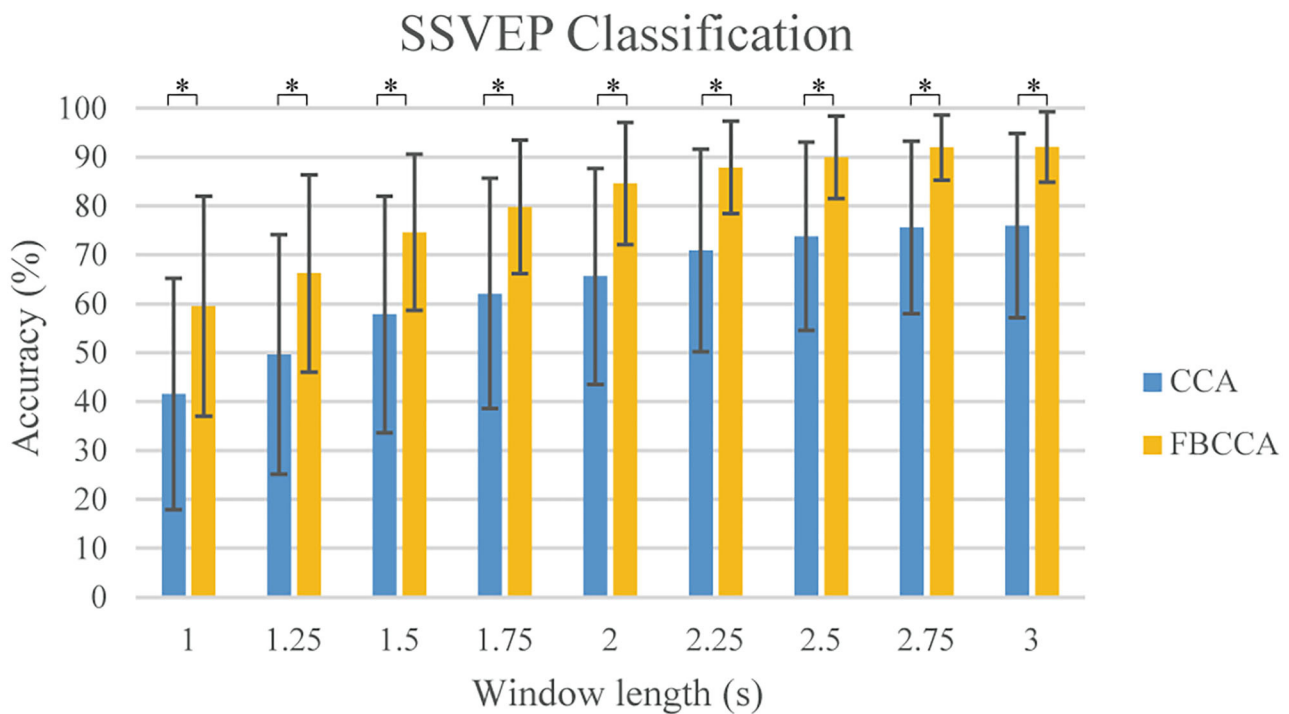
FIGURE 9 | The averaged classification accuracy for 11 subjects under different window length in the cue-based experiment. The blue bar shows the results of CCA and the yellow bar shows the results of FBCCA. Error bars are standard deviations. The asterisk indicates 1% significance level between CCA and FBCCA methods (t-test).

FIGURE 10 | Individual classification accuracy of CCA and FBCCA in the cue-based experiment. Nine bars in a subject indicated nine window length from 1 s **(Left)** to 3 s **(Right)** at a step of 0.25 s. The blue bar shows the results of CCA and the yellow bar shows the results of FBCCA.
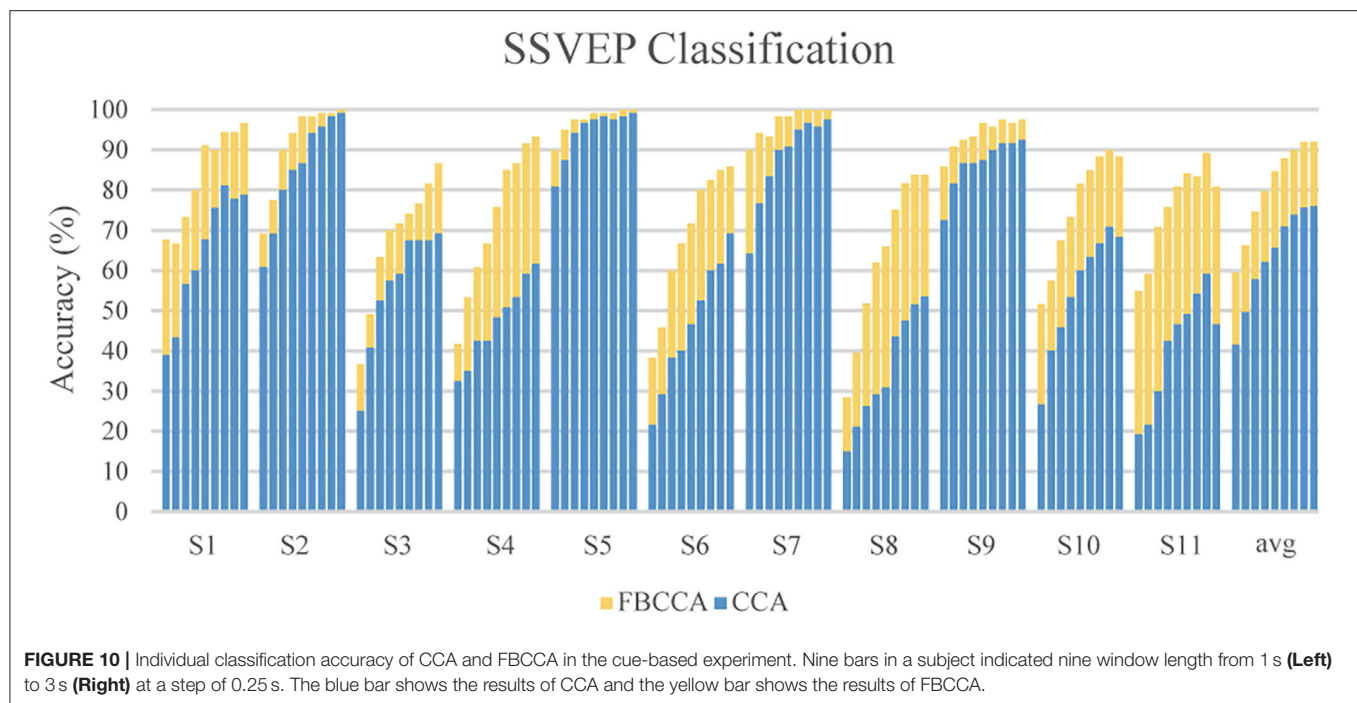
of commands executed by the robotic arm. Therefore, the wink-based cancel command helped to improve the effectiveness of robotic arm control.

As for the classification of the SSVEP-based BCI, we used the FBCCA method for classification, and we compared the classification results of the FBCCA and CCA methods in different window length which was shown in **Figure 9**. The statistical results revealed that the classification accuracy of the FBCCA was significantly better than the results of the CCA at each window lengths ($p < 0.01$). For the same length of stimulation time, the classification accuracy of FBCCA is better than CCA due to the use of harmonic frequencies information (Chen et al., 2015a,b). **Figure 10** showed the individual classification accuracy of CCA and FBCCA methods in the cue-based experiment. The results showed that FBCCA outperformed CCA in each window length for all subjects, especially for those subjects with lower classification accuracy in CCA. However, the flash of buttons in GUI is still easy to cause user fatigue. Recently, several studies attempted to flash the buttons at high frequency to reduce user fatigue (Allison et al., 2010; Diez et al., 2011). Furthermore, several studies showed that BCI combined with technologies like computer vision and deep learning also improve the performance of BCI and reduce the workload of users (Tayeb et al., 2018; Chen et al., 2019). The combination of the proposed asynchronous hybrid BCI and new technologies as future research direction will make the BCI more convenience and user friendly.

CONCLUSION

This paper proposed a hybrid BCI which combined SSVEP-based BCI and an EOG-based switch for asynchronous control of the

robotic arm. To decrease the FPR in asynchronous BCI, we designed the EOG-based switch to turn on and off the stimulus and a cancel command to effectively accomplish complex tasks. Two online experiments verified the feasibility of the subjects to use the EOG-based switch by a triple blink to activate or deactivate the flash of the SSVEP-based BCI which was used to select the control commands for the operation of the robotic arm to complete a series of complicated movements. And subjects were allowed to timely cancel the current command in feedback phase for more effective control of the robotic arm. All subjects succeeded in asynchronously operating the robotic arm to grasp, lift, and move a target object from the initial position to a specific location. The experimental results suggested that effective combination of EOG and SSVEP signals was able to realize an asynchronous hybrid BCI which allowed user to directly communicate with the external environment based on their own intention.

DATA AVAILABILITY STATEMENT

The raw data supporting the conclusions of this article are available on request to the corresponding author.

ETHICS STATEMENT

The studies involving human participants were reviewed and approved by Human Subjects Institutional Review Board of Huazhong University of Science and Technology, China. The patients/participants provided their written informed consent to participate in this study. Written informed consent was obtained from the individual(s) for the publication

of any potentially identifiable images or data included in this article.

AUTHOR CONTRIBUTIONS

YZ, JL, and PL proposed the idea and designed the experiments. YZ conducted the experiments. YZ, YL analyzed the data. YZ and PL wrote the manuscript. JL and PL provided facilities and equipment. All authors contributed to the article and approved the submitted version.

REFERENCES

Allison, B., Luth, T., Valbuena, D., Teymourian, A., Volosyak, I., and Graser, A. (2010). BCI demographics: how many (and what kinds of) people can use an SSVEP BCI? *IEEE Trans. Neural Syst. Rehabil. Eng.* 18, 107–116. doi: 10.1109/TNSRE.2009.2039495

Bin, G., Gao, X., Yan, Z., Hong, B., and Gao, S. (2009). An online multi-channel SSVEP-based brain–computer interface using a canonical correlation analysis method. *J. Neural Eng.* 6:46002. doi: 10.1088/1741-2560/6/4/046002

Brainard, D. H. (1997). The psychophysics toolbox. *Spat. Vis.* 10, 433–436. doi: 10.1163/156856897X00357

Chen, X., Chen, Z., Gao, S., and Gao, X. (2014). A high-ITR SSVEP-based BCI speller. *Brain-Computer Interfaces* 1, 181–191. doi: 10.1080/2326263X.2014.944469

Chen, X., Wang, Y., Gao, S., Jung, T.-P., and Gao, X. (2015a). Filter bank canonical correlation analysis for implementing a high-speed SSVEP-based brain–computer interface. *J. Neural Eng.* 12:46008. doi: 10.1088/1741-2560/12/4/046008

Chen, X., Wang, Y., Nakanishi, M., Gao, X., Jung, T.-P., and Gao, S. (2015b). High-speed spelling with a noninvasive brain–computer interface. *Proc. Natl. Acad. Sci.* 112, E6058–E6067. doi: 10.1073/pnas.1508080112

Chen, X., Wang, Y., Zhang, S., Gao, S., Hu, Y., and Gao, X. (2017). A novel stimulation method for multi-class SSVEP-BCI using intermodulation frequencies. *J. Neural Eng.* 14:26013. doi: 10.1088/1741-2552/aa5989

Chen, X., Zhao, B., Wang, Y., and Gao, X. (2019). Combination of high-frequency SSVEP-based BCI and computer vision for controlling a robotic arm. *J. Neural Eng.* 16:26012. doi: 10.1088/1741-2552/aaf594

Chen, X., Zhao, B., Wang, Y., Xu, S., and Gao, X. (2018). Control of a 7-DOF Robotic Arm System With an SSVEP-Based BCI. *Int. J. Neural Syst.* 28:1850018. doi: 10.1142/S0129065718500181

Cheng, M., Gao, X., Gao, S., and Xu, D. (2002). Design and implementation of a brain-computer interface with high transfer rates. *IEEE Trans. Biomed. Eng.* 49, 1181–1186. doi: 10.1109/TBME.2002.803536

Diez, P. F., Mut, V. A., Avila Perona, E. M., and Laciar Leber, E. (2011). Asynchronous BCI control using high-frequency SSVEP. *J. Neuroeng. Rehabil.* 8:39. doi: 10.1186/1743-0003-8-39

Farwell, L. A., and Donchin, E. (1988). Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials. *Electroencephalogr. Clin. Neurophysiol.* 70, 510–523. doi: 10.1016/0013-4694(88)90149-6

Floriano, A., F., Diez, P., and Freire Bastos-Filho, T. (2018). Evaluating the influence of chromatic and luminance stimuli on SSVEPs from behind-the-ears and occipital areas. *Sensors* 18:615. doi: 10.3390/s18020615

Gao, Q., Dou, L., Belkacem, A. N., and Chen, C. (2017). Noninvasive electroencephalogram based control of a robotic arm for writing task using hybrid BCI system. *Biomed Res. Int.* 2017, 1–8. doi: 10.1155/2017/8316485

He, S., Yu, T., Gu, Z., and Li, Y. (2017). "A hybrid BCI web browser based on EEG and EOG signals," in *2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)* (Seogwipo: IEEE), 1006–1009. doi: 10.1109/EMBC.2017.8036996

Herrmann, C. S. (2001). Human EEG responses to 1-100 Hz flicker: resonance phenomena in visual cortex and their potential correlation to cognitive phenomena. *Exp. brain Res.* 137, 346–353. doi: 10.1007/s002210100682

Huang, Q., Zhang, Z., Yu, T., He, S., and Li, Y. (2019). An EEG-/EOG-based hybrid brain-computer interface: application on controlling an integrated wheelchair robotic arm system. *Front. Neurosci.* 13:1243. doi: 10.3389/fnins.2019.01243

Khan, M. J., and Hong, K.-S. (2017). Hybrid EEG-fNIRS-based eight-command decoding for BCI: application to quadcopter control. *Front. Neurorobot.* 11:6. doi: 10.3389/fnbot.2017.00006

Li, Y., He, S., Huang, Q., Gu, Z., and Yu, Z. L. (2018). A EOG-based switch and its application for "start/stop" control of a wheelchair. *Neurocomputing* 275, 1350–1357. doi: 10.1016/j.neucom.2017.09.085

Ma, J., Zhang, Y., Cichocki, A., and Matsuno, F. (2015). A novel EOG/EEG hybrid human–machine interface adopting eye movements and ERPs: application to robot control. *IEEE Trans. Biomed. Eng.* 62, 876–889. doi: 10.1109/TBME.2014.2369483

Manyakov, N. V., Chumerin, N., Robben, A., Combaz, A., van Vliet, M., and Van Hulle, M. M. (2013). Sampled sinusoidal stimulation profile and multichannel fuzzy logic classification for monitor-based phase-coded SSVEP brain–computer interfacing. *J. Neural Eng.* 10:36011. doi: 10.1088/1741-2560/10/3/036011

Nakanishi, M., and Mitsukura, Y. (2013). "Wheelchair control system by using electrooculogram signal processing," in *The 19th Korea-Japan Joint Workshop on Frontiers of Computer Vision* (Incheon: IEEE), 137–142. doi: 10.1109/FCV.2013.6485476

Ortner, R., Allison, B. Z., Korisek, G., Gaggl, H., and Pfurtscheller, G. (2011). An SSVEP BCI to control a_ hand orthosis for persons with tetraplegia. *IEEE Trans. Neural Syst. Rehabil. Eng.* 19, 1–5. doi: 10.1109/TNSRE.2010.2076364

Pan, J., Li, Y., Zhang, R., Gu, Z., and Li, F. (2013). Discrimination between control and idle states in asynchronous SSVEP-based brain switches: a pseudo-key-based approach. *IEEE Trans. Neural Syst. Rehabil. Eng.* 21, 435–443. doi: 10.1109/TNSRE.2013.2253801

Pfurtscheller, G., Allison, B. Z., Brunner, C., Bauernfeind, G., Solis-Escalante, T., Scherer, R., et al. (2010a). The hybrid BCI. *Front. Neurosci.* 4:30. doi: 10.3389/fnpro.2010.00003

Pfurtscheller, G., Solis-escalante, T., Member, S., Ortner, R., Linortner, P., and Müller-putz, G. R. (2010b). Self-paced Operation of an SSVEP-based orthosis with and without an imagery-based "brain switch:" a feasibility study towards a hybrid *BCI.* 18, 409–414. doi: 10.1109/TNSRE.2010.2040837

Takano, K., Komatsu, T., Hata, N., Nakajima, Y., and Kansaku, K. (2009). Visual stimuli for the P300 brain-computer interface: a comparison of white/gray and green/blue flicker matrices. *Clin. Neurophysiol.* 120, 1562–1566. doi: 10.1016/j.clinph.2009.06.002

Tan, C., Sun, F., Fang, B., Kong, T., and Zhang, W. (2019). Autoencoder-based transfer learning in brain–computer interface for rehabilitation robot. *Int. J. Adv. Robot. Syst.* 16, 1–12. doi: 10.1177/1729881419840860

Tayeb, Z., Waniek, N., Fedjaev, J., Ghaboosi, N., Rychly, L., Widderich, C., et al. (2018). Gumpy: a Python toolbox suitable for hybrid brain–computer interfaces. *J. Neural Eng.* 15:65003. doi: 10.1088/1741-2552/aae186

Tomita, Y., Vialatte, F.-B., Dreyfus, G., Mitsukura, Y., Bakardjian, H., and Cichocki, A. (2014). Bimodal BCI using simultaneously NIRS and EEG. *IEEE Trans. Biomed. Eng.* 61, 1274–1284. doi: 10.1109/TBME.2014.2300492

Wang, H., Li, Y., Long, J., Yu, T., and Gu, Z. (2014). An asynchronous wheelchair control by hybrid EEG–EOG brain–computer interface. *Cogn. Neurodyn.* 8, 399–409. doi: 10.1007/s11571-014-9296-y

Wolpaw, J. R., Birbaumer, N., Heetderks, W. J., McFarland, D. J., Peckham, P. H., Schalk, G., et al. (2000). Brain-computer interface technology: a review of the first international meeting. *IEEE Trans. Rehabil. Eng.* 8, 164–173. doi: 10.1109/TRE.2000.847807

Wolpaw, J. R., Birbaumer, N., McFarland, D. J., Pfurtscheller, G., and Vaughan, T. M. (2002). Brain–computer interfaces for communication and control. *Clin. Neurophysiol.* 113, 767–791. doi: 10.1016/S1388-2457(02)00057-3

Wolpaw, J. R., McFarland, D. J., Neat, G. W., and Forneris, C. A. (1991). An EEG-based brain-computer interface for cursor control. *Electroencephalogr. Clin. Neurophysiol.* 78, 252–259. doi: 10.1016/0013-4694(91)90040-B

Zhou, Y., He, S., Huang, Q., and Li, Y. (2020). A hybrid asynchronous brain-computer interface combining SSVEP and EOG signals. *IEEE Trans. Biomed. Eng.* 67, 2881–2892. doi: 10.1109/TBME.2020.2972747

A New Projected Active Set Conjugate Gradient Approach for Taylor-Type Model Predictive Control: Application to Lower Limb Rehabilitation Robots With Passive and Active Rehabilitation

Tian Shi[1], Yantao Tian[1], Zhongbo Sun[2,3*], Bangcheng Zhang[4], Zaixiang Pang[4], Junzhi Yu[5*] and Xin Zhang[2]

[1] College of Communication Engineering, Jilin University, Changchun, China, [2] Department of Control Engineering, Changchun University of Technology, Changchun, China, [3] Key Laboratory of Bionic Engineering of Ministry of Education, Jilin University, Changchun, China, [4] School of Mechatronical Engineering, Changchun University of Technology, Changchun, China, [5] State Key Laboratory for Turbulence and Complex Systems, Department of Advanced Manufacturing and Robotics, College of Engineering, Peking University, Beijing, China

In this paper, a three-order Taylor-type numerical differentiation formula is firstly utilized to linearize and discretize constrained conditions of model predictive control (MPC), which can be generalized from lower limb rehabilitation robots. Meanwhile, a new numerical approach that projected an active set conjugate gradient approach is proposed, analyzed, and investigated to solve MPC. This numerical approach not only incorporates both the active set and conjugate gradient approach but also utilizes a projective operator, which can guarantee that the equality constraints are always satisfied. Furthermore, rigorous proof of feasibility and global convergence also shows that the proposed approach can effectively solve MPC with equality and bound constraints. Finally, an echo state network (ESN) is established in simulations to realize intention recognition for human–machine interactive control and active rehabilitation training of lower-limb rehabilitation robots; simulation results are also reported and analyzed to substantiate that ESN can accurately identify motion intention, and the projected active set conjugate gradient approach is feasible and effective for lower-limb rehabilitation robot of MPC with passive and active rehabilitation training. This approach also ensures computational when disturbed by uncertainties in system.

Keywords: rehabilitation robot, model predictive control, intention recognition, conjugate gradient approach, projected operator

1. INTRODUCTION

The number of limb impairment patients who were injured by stroke has increased year by year, and this disease has also been developing in the direction of youth, seriously endangering the health of patients (Zorowitz et al., 2013). Recently, researchers have paid a great deal of attention to robotics to promote the development of scientific and engineering fields (Jin et al., 2017; Jin and Li, 2018; Xie et al., 2019, 2020; Zhang et al., 2020). Compared to traditional rehabilitation training methods

that see problems like resource consumption, high costs, and long rehabilitation period, lower-limb rehabilitation robots can be deemed as a more effective method for recovering patients' movement function. In virtue of interaction generally existing between the lower-limb rehabilitation robot and the patient, to avoid the a second injury in the patient during rehabilitation training, it is essential to propose a human–machine interactive control method, which can be utilized to investigate and analyze the lower-limb rehabilitation robot (Fleischer and Hommel, 2008).

Intention recognition is one of the key points for realizing human–machine interactive control methods with lower-limb rehabilitation robots. Generally, motion intentions include joint angles and angular velocities, which can be recognized by decoding bioelectrical signals. Afterwards, the intentions are referenced by the lower-limb rehabilitation robot to complete interaction (Ding et al., 2016; Peng et al., 2018). An appropriate alternative is to establish a relationship between biological signals and movements for the patient. The surface electromyography (sEMG) can be regarded as the biological signals, which is a micro-electrical signal that appears 20–80 ms before the muscle contraction (Fleischer and Hommel, 2008). Involving two approaches to construct the relationship, one of which is a physiological muscle model, such as the Hill muscle model and the Hammerstein muscle model (Hunt et al., 1998; Buchanan et al., 2004), muscle forces and joint motions can be estimated by those models from sEMG signals; meanwhile, the unidentified physiological parameters of those models affect their applications in rehabilitation systems (Han et al., 2015). Another is the regression model, which can be established to connect sEMG signals to indicate intentions in a straightforward manner, rather than considering physiological parameters (Ding et al., 2016). For instance, a BP neural network was exploited to describe the relationship between sEMG signals and motion intentions, which was verified on able-bodied subjects and patients (Zhang et al., 2012); least squares support vector regression was proposed to predict periodic lower-limb motions from multi-channel sEMG signals (Li et al., 2015). Those approaches constitute a connection between human's bioelectrical signals and motion intentions, and intention recognition can thus be realized effectively.

During the rehabilitation training, the motion trajectory, which is a predetermined curve or recognized by sEMG signals, is known to the rehabilitation device, and the patient is assisted by the lower-limb rehabilitation robot to recover. However, it is very important to avoid the risk of second injury for patients in rehabilitation, and a human–machine interactive control method should therefore be considered to increase security and stability of rehabilitation robots. Recently, some classical control methods have been developed and applied to rehabilitation robots, for example, the rehabilitative system was realized by an adaptive control framework and a human–machine interactive method; meanwhile, the potential conflicts between patient and rehabilitation robot were rejected by position-dependent stiffness and predetermined trajectory (Zhang and Cheah, 2015). Pehlivan et al. (2016) presented a minimal assist-as-needed controller for rehabilitation robots, which could provide corresponding assistance for patients during rehabilitation training.

In recent years, model predictive control (MPC) not only considered the constraints of a non-linear system but predicted future states. People have therefore paid more attention to investigating it and applying it to the aerospace, automobile, economics, and robotics fields. As the patient should be assisted by the lower-limb rehabilitation robot within a relatively safe motion range and protected against accidents, the MPC method is suitable for human–machine interactive control of the lower-limb rehabilitation robot. Generally speaking, the MPC method is designed to optimize multivariable and constrained control systems. On the one hand, a control sequence is created by minimizing an optimization objective function over a finite prediction horizon within state and control constraints. On the other hand, the first optimal solution of non-linear optimization problem feeds back to the non-linear systems, which is utilized to generate the next iteration (Mayne, 2014). A key issue for MPC is that the computational burden of real-time optimization should be reduced through neural networks (Yan and Wang, 2014). To solve this problem is to linearize non-linear systems and discretize the differential term. More and more people have consequently developed some classical methods. For example, a neural network was utilized to identify the unknown non-linear discrete system, and then one-order Taylor expansion formula was used to linearize the MPC problem (Pan and Wang, 2012). For non-linear continuous systems, in order to discretize the differential term and guarantee the higher precision, a Taylor-type numerical differentiation formula was developed and applied to solve non-linear time-varying optimization problem (Zhang et al., 2019), non-linear time-varying equations (Jin et al., 2019), time-varying matrix inversion (Guo et al., 2017), future dynamic non-linear optimization problem (Wei et al., 2019), time-dependent Sylvester equations (Qi et al., 2020), and so on. As the MPC problem can be seen as an optimization problem, a Taylor-type numerical differentiation formula is thus also suitable for solving the MPC problem online.

Another key issue of MPC to be looked at further is online optimization. In fact, an MPC problem can be converted to a non-linear optimization problem with equality constraints and bound constraints and be solved to obtain the control sequence at each sample time. There are thus numerous algorithms proposed and studied for this non-linear optimization problem. A complex-valued discrete-time neural dynamics is studied by Qi et al. (2019) for solving time-dependent complex quadratic programming (QP), which possesses high accuracy and strong robustness but is only suitable for linear constrained QP problems. A trust region-sequential quadratic programming approach attempts to solve a sequence of QP subproblems of non-linear constraint optimization problems, which is based on trust-region technology and applied by finding the Karush-Kuhn-Tucker (KKT) points. It is noteworthy for the approach that the compatibility of the QP subproblem and the Maratos effect were overcome by adding several linear equations into the traditional trust region-SQP algorithm (Sun et al., 2019). However, the calculation of this algorithm is increased by the added linear equations. In addition, some troubles maybe emerge; for instance, one is the consistency of the coefficient matrix and the other the computational burden. Similarly,

conjugate gradient methods can also be regarded as an effective optimization approach that utilizes an iteration point with a steep descent direction to generate conjugate direction and compute a global minimum point instead of solving linear equations of trust region-SQP algorithm (Sun et al., 2019). Some classical conjugate gradient methods include the Hestenes-Stiefel (HS) method (Hestenes and Stiefel, 1952), the Fletcher-Reeves (FR) method (Fletcher and Reeves, 1964), the Polak-Ribiére-Polyak (PRP) method (Polak and Ribière, 1969; Polyak, 1969), the Dai-Yuan (DY) method (Dai and Yuan, 2000a), the Liu-Storey (LS) method (Liu and Storey, 1991), and the conjugate descent (CD) method (Dai and Yuan, 2000b), but those methods were exploited to solve unconstrained optimization problems off-line. The MPC problem usually contains some constraint qualifications. Some modified conjugate gradient methods, which consider the constrained conditions, were thus proposed by modifying the search direction and a projected operator (Dai, 2014; Sun et al., 2018). Besides, a projected gradient method, which projected the gradient into the feasible region, was proposed by Rosen (1960), and some modified conjugate gradient methods were extended by some researchers based on the mentioned methods (Li and Li, 2013; Dai, 2014). Those modified conjugate gradient methods were also applied in optimal robust controllers and robots (Sun et al., 2018). In this paper, a modified conjugate gradient method, which will simultaneously consider equality constraints and bound constraints, will be utilized to solve MPC problem. Furthermore, the proposed algorithm of this paper is further applied to the lower limb rehabilitation robots with passive and active rehabilitation.

There are three significant contributions to be developed in this paper. The primary one is that a new projected active set conjugate gradient approach is developed and investigated, and rigorous proof of feasibility and global convergence is also given. The second is that a relationship between sEMG signals and motion intentions established by an echo state network (ESN) model can identify human motion state. Finally, a numerical simulation about passive and active rehabilitation training of lower-limb rehabilitation robot is illustrated and solved by the proposed method. Surprisingly, the studies on the rehabilitation training of lower-limb rehabilitation robot for MPC problem with projected active set conjugate gradient approach are scarce. This motivates our present study.

The rest of this paper is organized as follows: In section 2, the MPC problem is introduced, and a three-order Taylor-type numerical differentiation formula is proposed and utilized to discretize MPC model. In section 3, a new projected active set conjugate gradient algorithm is developed, analyzed, and investigated for the MPC problem, which can be generalized from non-linear systems and lower limb rehabilitation robots. Furthermore, the feasibility and the global convergence of this approach are also proven. The relationship of sEMG signals and motion intentions is established by an ESN model in section 5; meanwhile, passive and active rehabilitation training of lower-limb rehabilitation robot is illustrated and simulated by the proposed method, which is based on sEMG signals with ESN model and MPC problem. Furthermore, the disturbance of dynamic model is also considered through simulation in

section 4. Finally, section 6 summarizes the results of lower-limb rehabilitation robot based on the MPC technique and expects future work.

2. FROM MPC TO NON-LINEAR CONSTRAINED OPTIMIZATION

2.1. Problem Description

In this subsection, consider the following non-linear control system:

$$\begin{cases} \dot{\mathbf{x}}^k = \mathbf{A}(\mathbf{x}^k)\mathbf{x}^k + \mathbf{B}(\mathbf{x}^k)\mathbf{u}^k + \mathbf{C}(\mathbf{x}^k), \\ \mathbf{y}^k = \mathbf{h}(\mathbf{x}^k), \end{cases} \quad (1)$$

where $\mathbf{x}^k \in R^n$ is a system state variable, $\mathbf{u}^k \in R^m$ is a control input signal, $\mathbf{A}(\mathbf{x}^k) \in R^{n \times n}$, $\mathbf{B}(\mathbf{x}^k) \in R^{n \times m}$, $\mathbf{C}(\mathbf{x}^k) \in R^n$ are the state matrix, control input matrix and constant matrix, respectively. $\mathbf{y}^k \in R^n$ denotes the system output, and $\mathbf{h}(\cdot)$ is a non-linear function.

MPC for the non-linear control system is devoted to generating a sequence of control signals by minimizing an objective function repeatedly over a finite moving prediction horizon with system state and input constraints satisfied simultaneously. For the non-linear control system (1), the MPC problem can be described as a non-linear discrete-time optimal control problem within input and state constraints:

$$\min_{\mathbb{X}^k, \mathbb{U}^k} \sum_{i=1}^{N} \left\| \mathbf{r}(k+i|k) - \mathbf{y}(k+i|k) \right\|_Q^2 + \sum_{j=0}^{N_u-1} \left\| \Delta \mathbf{u}(k+j|k) \right\|_R^2$$

$$\text{s.t. } \dot{\mathbf{x}}(k+i|k) = \mathbf{A}(k+i|k)\mathbf{x}(k+i|k) + \mathbf{B}(k+i|k)\mathbf{u}(k+j|k)$$
$$+ \mathbf{C}(k+i|k),$$
$$\mathbf{y}(k+i|k) = \mathbf{h}(\mathbf{x}(k+i|k)),$$
$$\mathbf{x}(k+i|k) \in [\mathbf{x}_{\min}, \mathbf{x}_{\max}], \mathbf{u}(k+j|k) \in [\mathbf{u}_{\min}, \mathbf{u}_{\max}],$$
$$i = 1, 2, \dots, N, j = 1, 2, \dots, N_u,$$

$$(2)$$

where $\mathbf{r}(k+i|k)$ and $\mathbf{y}(k+i|k)$ are the desired output and the predicted output for ith step ahead from kth sampling instant; $\Delta \mathbf{u}(k+j|k) = \mathbf{u}(k+j|k) - \mathbf{u}(k+j-1|k)$ denotes the control increment; N and N_u are the prediction horizon and control horizon, respectively; $\mathbb{X}^k = \{\mathbf{x}(k+1|k), \mathbf{x}(k+2|k), \dots, \mathbf{x}(k+N|k)\}$, $\mathbb{U}^k = \{\mathbf{u}(k|k), \mathbf{u}(k+1|k), \dots, \mathbf{u}(k+N_u-1|k)\}$; $Q \in R^{n \times n}$ and $R \in R^{m \times m}$ are real positive definite matrix, $\|\cdot\|_Q$ and $\|\cdot\|_R$ are Euclidean norms defined as $\|x\|_{\mathfrak{A}} = \sqrt{x^T \mathfrak{A} x}$, where $\mathfrak{A}$ is a square matrix.

2.2. Three-Order Taylor-Type Discretization for MPC

In this subsection, a three-order Taylor-type numerical differentiation formula with truncation error of $\mathbf{O}(h^2)$ is constructed for the first-order derivative approximation and is exploited to discretize MPC problem (Jin and Zhang, 2014). To obtain the higher-order truncation error, Guo et al. (2017) proposed a novel Taylor-type numerical differentiation formula for the time-varying matrix inversion.

Lemma 1. Assume that $\mathbf{x}^k \in C^4[a, b]$ and $\mathbf{x}^{k-4}$, $\mathbf{x}^{k-3}$, $\mathbf{x}^{k-2}$, $\mathbf{x}^{k-1}$, $\mathbf{x}^k$, $\mathbf{x}^{k+1} \in [a, b]$, h denotes the sampling gap. Subsequently, a three-order Taylor-type numerical differentiation formula can be obtained as follows:

$$\dot{\mathbf{x}}^k \approx \frac{13\mathbf{x}^{k+1} - 7\mathbf{x}^k + 2\mathbf{x}^{k-1} - 10\mathbf{x}^{k-2} + \mathbf{x}^{k-3} + \mathbf{x}^{k-4}}{24h} \quad (3)$$

with a truncation error of $\mathbf{O}(h^3)$.

Proof. See Guo et al. (2017).

According to the Lemma 1, the non-linear control system (1) can be discreted as

$$\begin{cases} \mathbf{x}^{k+1} = (\frac{24}{13}h\mathbf{G}^k + \frac{7}{13}I)\mathbf{x}^k - \frac{2}{13}\mathbf{x}^{k-1} + \frac{10}{13}\mathbf{x}^{k-2} - \frac{1}{13}\mathbf{x}^{k-3} \\ \quad\quad - \frac{1}{13}\mathbf{x}^{k-4} + \frac{24}{13}h\mathbf{B}^k\mathbf{u}^k + \frac{24}{13}h\mathbf{Z}^k \\ \mathbf{y}^k = \mathbf{h}(\mathbf{x}^k), \end{cases} \quad (4)$$

where $\mathbf{f} = \mathbf{A}(\mathbf{x})\mathbf{x} + \mathbf{B}(\mathbf{x})\mathbf{u} + \mathbf{C}(\mathbf{x})$, $\mathbf{G}^k = \frac{\partial \mathbf{f}}{\partial \mathbf{x}}(\mathbf{x}^k, \mathbf{u}^k)$, $\mathbf{B}^k = \mathbf{B}(\mathbf{x}^k) = \frac{\partial \mathbf{f}}{\partial \mathbf{u}}(\mathbf{x}^k, \mathbf{u}^k)$, $\mathbf{Z}^k = \mathbf{f}(\mathbf{x}^k, \mathbf{u}^k) - \mathbf{G}^k\mathbf{x}^k - \mathbf{B}^k\mathbf{u}^k$, I is an identity matrix.

Therefore, the MPC problem (2) can be rewritten in the following form:

$$\min_{\mathbb{X}^k, \mathbb{U}^k} \quad \sum_{i=1}^{N} \left\| \mathbf{r}(k+i|k) - \mathbf{y}(k+i|k) \right\|_Q^2 + \sum_{j=0}^{N_u-1} \left\| \Delta\mathbf{u}(k+j|k) \right\|_R^2$$

$$\text{s.t.} \quad \mathbf{x}^{i+1} = (\frac{24}{13}h\mathbf{G}^i + \frac{7}{13}I)\mathbf{x}^i - \frac{2}{13}\mathbf{x}^{i-1} + \frac{10}{13}\mathbf{x}^{i-2} - \frac{1}{13}\mathbf{x}^{i-3}$$

$$\quad\quad - \frac{1}{13}\mathbf{x}^{i-4} + \frac{24}{13}h\mathbf{B}^i\mathbf{u}^j + \frac{24}{13}h\mathbf{Z}^i,$$

$$\mathbf{y}^i = \mathbf{h}(\mathbf{x}^i),$$

$$\mathbf{x}^i \in [\mathbf{x}_{\min}, \mathbf{x}_{\max}], \mathbf{u}^j \in [\mathbf{u}_{\min}, \mathbf{u}_{\max}],$$

$$i = 1, 2, \ldots, N, j = 1, 2, \ldots, N_u,$$

$$(5)$$

where $\mathbf{x}^i = \mathbf{x}(k+i|k)$, $\mathbf{u}^j = \mathbf{u}(k+j|k)$, and $\mathbf{y}^i = \mathbf{y}(k+i|k)$, and another symbols see **Appendix**.

In general, the MPC problem can be solved by the conjugate gradient approach (Šantin and Havlena, 2011), and a sequence

$$\mathbf{u}^{(i)} := [u_0^{(i)}; u_1^{(i)}; \ldots; u_{N_u-1}^{(i)}]$$

can be regarded as an initial control input sequence of the MPC problem at Step i. In addition, the first element u_0^* of optimal solution $\mathbf{u}^*$ can be seen as a feedback control law for non-linear control system (1).

For simplicity, the MPC problem (5) is equivalent to the following non-linear optimization problem with linear equality constrain and bound constrain:

$$\min_x \quad \Gamma(x)$$

$$\text{s.t.} \quad \Lambda x = b, \ x \in \Omega, \quad (6)$$

where $x = (\mathbf{x}, \mathbf{u})$, $\Gamma(\cdot)$ is a continuously differentiable function, Λ is a constant matrix, b is a constant vector, and $\Omega = \{x = (\mathbf{x}, \mathbf{u}) | \mathbf{x}_{\min} \leq \mathbf{x} \leq \mathbf{x}_{\max}, \mathbf{u}_{\min} \leq \mathbf{u} \leq \mathbf{u}_{\max}\}$ is a bound constrained set.

3. PROJECTED ACTIVE SET CONJUGATE GRADIENT ALGORITHM FOR NON-LINEAR CONSTRAINED OPTIMIZATION

In this section, a projected active set conjugate gradient algorithm is proposed to solve the following non-linear optimization problem:

$$\min_x \quad \Gamma(x)$$

$$\text{s.t.} \quad \Lambda x = b, \quad (7)$$

$$x \in \Omega = \{x | s \leq x \leq t\},$$

and the convergence of the proposed approach is developed, investigated, and analyzed as follows.

3.1. Projected Active Set HS-Type Conjugate Gradient Algorithm

To further analyze projected active set conjugate gradient algorithm, some basic definitions and notions should be revisited in this subsection. Let x^* be a stationary point of (7), and consider the following active set:

$$H^* = \{i : x_i^* = s_i \text{ or } x_i^* = t_i\}.$$

Furthermore, define

$$L^* = \{1, 2, \ldots, n\} \setminus H^*,$$

as a set of free variables, where L^* is the complement of H^*. Therefore, the KKT conditions for the problem (7) can be converted as follows:

$$\begin{cases} (s_i + t_i - 2x_i^*)\nabla\Gamma_i(x^*) \geq 0, & \text{if } i \in H^*, \\ \nabla\Gamma_i(x^*) = 0, & \text{if } i \in L^*, \end{cases}$$

where $\nabla\Gamma_i(x)$ is the ith element of the gradient for Γ at x. According to the literature (Kanzow and Klug, 2006), $H(x)$ and $L(x)$, which approximate the active set and the free variables set can be defined as follows:

$$H(x) = \{i : s_i \leq x_i \leq s_i + \psi(x) \text{ or } t_i - \psi(x) \leq x_i \leq t_i\},$$

$$L(x) = \{i : s_i + \psi(x) < x_i < t_i - \psi(x)\},$$

where $\psi(x) = \min\{\xi(x), \psi_0\}$ and $\xi(x) = \sqrt{\left\| x - P_\Omega(x - \nabla\Gamma(x)) \right\|}$, $P_\Omega(\cdot)$ is a projection function defined as $P_\Omega(\mathbf{x}) = \operatorname{argmin}_{\omega \in \Omega} \|\mathbf{x} - \omega\|_2$. ψ_0 is a positive scalar, which should be sufficiently small. Furthermore, it should satisfy the following inequality:

$$0 < \psi_0 < \min_{i=1,2,\ldots,n} \frac{1}{3}(s_i - t_i).$$

In what follows, let x^k be the point of iteration k, and for simplicity, we abbreviate $H(x^k)$ and $L(x^k)$ as H^k and L^k.

According to the literature (Cheng et al., 2014), the active set H^k will be divided into the following three parts:

$$H_1^k = \{i : x_i^k = s_i \text{ or } x_i^k = t_i, \text{ and } (s_i + t_i - 2x_i^k)\nabla\Gamma_i^k \geq 0\},$$
$$H_2^k = \{i : s_i \leq x_i^k \leq s_i + \psi(x^k) \text{ or } t_i - \psi(x^k) \leq x_i^k \leq t_i, \text{and}$$
$$(s_i + t_i - 2x_i^k)\nabla\Gamma_i^k < 0\},$$
$$H_3^k = \{i : s_i < x_i^k \leq s_i + \psi(x^k) \text{ or } t_i - \psi(x^k) \leq x_i^k < t_i, \text{and}$$
$$(s_i + t_i - 2x_i^k)\nabla\Gamma_i^k \geq 0\}.$$
$$(8)$$

It is inferred that a search direction d^k can be constructed as a feasible direction of Γ at x^k if and only if $d_i^k \geq 0$, $i \in \{i : x_i^k = s_i \text{ and } \nabla\Gamma_i^k \geq 0\}$ and $d_i^k \leq 0$, $i \in \{i : x_i^k = t_i \text{ and } \nabla\Gamma_i^k \leq 0\}$.

It is demonstrated that the active set H_1^k can be seen as the equality constraints of (7), and according to Rosen's gradient projection method (Rosen, 1960; Dai, 2014), an active set projection matrix is given as follows:

$$\mathscr{P}^k = I^k - (M^k)^T(M^k(M^k)^T)^{-1}M^k,$$
$$M^k = \begin{bmatrix} E^k \\ \Lambda \end{bmatrix},$$
$$(9)$$

where E^k satisfies $E_{(L,\cdot)}^k x^k = s_L$ and $E_{(U,\cdot)}^k x^k = -t_U$, $L = \{i : x_i^k = s_i \text{ and } (s_i + t_i - 2x_i^k)\nabla\Gamma_i^k \geq 0\}$, $U = \{i : x_i^k = t_i \text{ and } (s_i + t_i - 2x_i^k)\nabla\Gamma_i^k \geq 0\}$,

Hence the search direction d^k is defined by

$$d^k = \begin{cases} -\mathscr{P}^k\nabla\Gamma^k, & \text{if } k = 0 \text{ or } \exists i \in H_1^k \cup H_2^k \cup H_3^k, \\ -\mathscr{P}^k\nabla\Gamma^k + \beta_k^{HS}d^{k-1} - \zeta_k\hat{z}^{k-1}, & \text{if } k \geq 1 \text{ and } \forall i \in L^k, \end{cases}$$
$$(10)$$

where

$$\beta_k^{HS} = \frac{(\mathscr{P}^k\nabla\Gamma^k)^T z^{k-1}}{(d^{k-1})^T v^{k-1}}, \zeta_k = \frac{(\nabla\Gamma^k)^T d^{k-1}}{(d^{k-1})^T v^{k-1}},$$
$$z^{k-1} = \nabla\Gamma^k - \nabla\Gamma^{k-1}, \hat{z}^{k-1} = \mathscr{P}^k z^{k-1},$$
$$v^{k-1} = z^{k-1} + \gamma w^{k-1}, \gamma = \gamma_0 + \max\{0, -\frac{(w^{k-1})^T z^{k-1}}{(w^{k-1})^T w^{k-1}}\},$$
$$w^{k-1} = x^k - x^{k-1},$$

and γ_0 is a positive constant. In what follows, the search direction d^k can be rigorously proved as a feasible descent direction of Γ at x^k for non-linear optimization problem (7).

Theorem 1. Suppose that $x^k \in \{x | \Lambda x = b, x \in \Omega\}$ holds, and x^k is not a stationary point of (7), d^k is defined by (10), then search direction d^k is a feasible descent direction of Γ at x^k for non-linear optimization problem (7).

Proof. According to (8) and (10), and as per definition of the projection matrix, the following two cases can be generalized.

Case 1. If $k = 0$ or $\exists i \in H_2^k \cup H_3^k$, then the inequality can be directly computed as follows:

$$(\nabla\Gamma^k)^T d^k = (\nabla\Gamma^k)^T(-\mathscr{P}^k\nabla\Gamma^k) = -\|\mathscr{P}\nabla\Gamma^k\|^2 \leq 0. \quad (11)$$

Case 2. If $k \geq 1$ and $\forall i \in L^k$, then the following inequality can be directly obtained:

$$(\nabla\Gamma^k)^T d^k = (\nabla\Gamma^k)^T(-\mathscr{P}^k\nabla\Gamma^k + \beta_k^{HS}d^{k-1} - \zeta_k\hat{z}^{k-1})$$
$$= -\|\mathscr{P}^k\nabla\Gamma^k\|^2 + \frac{(\mathscr{P}^k\nabla\Gamma^k)^T z^{k-1}}{(d^{k-1})^T v^{k-1}}(\nabla\Gamma^k)^T d^{k-1}$$
$$- \frac{(\nabla\Gamma^k)^T d^{k-1}}{(d^{k-1})^T v^{k-1}}(\nabla\Gamma^k)^T \mathscr{P}^k z^{k-1}$$
$$= -\|\mathscr{P}^k\nabla\Gamma^k\|^2 + (d^{k-1})^T\frac{\nabla\Gamma^k(\nabla\Gamma^k)^T \mathscr{P}^k z^{k-1}}{(d^{k-1})^T v^{k-1}}$$
$$- \frac{(d^{k-1})^T\nabla\Gamma^k}{(d^{k-1})^T v^{k-1}}(\nabla\Gamma^k)^T \mathscr{P}^k z^{k-1}$$
$$= -\|\mathscr{P}^k\nabla\Gamma^k\|^2 \leq 0.$$
$$(12)$$

The search direction d^k is therefore a descent direction of Γ at x^k.

Now, the proof of the feasibility for d^k is shown as follows, and it is further inferred that

$$M^k(\mathscr{P}^k\nabla\Gamma^k) = M^k(I - (M^k)^T(M^k(M^k)^T)^{-1}M^k)\nabla\Gamma^k = 0.$$

If E^k is not an empty set, it can be seen that

$$M^k d^k = M^k(-\mathscr{P}^k\nabla\Gamma^k) = 0. \quad (13)$$

If E^k is an empty set, then $M^k = \Lambda$. Owing to Equation (10), then the following equation can be generalized as

$$M^k d^k = M^k(-\mathscr{P}\nabla\Gamma^k + \beta_k^{HS}d^{k-1} - \zeta_k\hat{z}^{k-1})$$
$$= -\Lambda\mathscr{P}\nabla\Gamma^k + \beta_k^{HS}\Lambda d^{k-1} - \zeta_k\Lambda\mathscr{P}(\nabla\Gamma^k - \nabla\Gamma^{k-1})$$
$$= \beta_k^{HS}\Lambda d^{k-1}.$$
$$(14)$$

If $k = 0$ or $\exists i \in H_1^k \cup H_2^k \cup H_3^k$, then

$$\Lambda d^k = -\Lambda\mathscr{P}^k\nabla\Gamma^k = 0. \quad (15)$$

Owing to Equations (13)–(15), then we have $M^k d^k = 0$ for all $k \geq 0$. It is also inferred that d^k is a feasible descent direction for non-linear optimization problem (7). □

According to the above analysis and investigation, the projected active set HS-type conjugate gradient algorithm (PASHS) is developed, analyzed, and verified for non-linear optimization problem (7).

Algorithm 1. (PASHS)

Step 0. Initialize $x^0 \in \{x | \Lambda x = b, x \in \Omega\}$ and projection matrix $\mathscr{P}^0$, let $k = 0$ and positive constants $\varepsilon, \psi_0, \gamma_0, \eta_0, \delta, \rho < 1$.

Step 1. If $\left\| P_\Omega(x^k - \mathscr{P}^k\nabla\Gamma^k) - x^k \right\| \leq \varepsilon$ or $k > k_{\max}$, stop; else go to Step 2.

Step 2. Compute d^k by (10).

Step 3. Determine a stepsize $\eta_k = \max\{\eta_0\rho^j | j = 0, 1, 2, \ldots\}$ by Armijio-type line search rule:

$$\Gamma(x^k + \eta_k d^k) \leq \Gamma(x^k) + \delta\eta_k(\mathscr{P}^k\nabla\Gamma^k)^T d^k. \quad (16)$$

Step 4. Let $x^{k+1} = P_\Omega(x^k + \eta_k d^k)$, and $k := k + 1$, go to Step 1.

Remark 1. A projected matrix $\mathscr{P}^k$ is computed by an active set, which ensures iteration point satisfying equality and bounded constraints of non-linear optimization problem (7). Assume that if the component the active set H^k is not contained in the previous iteration point x^k, the search direction d^k is updated by the second formula of (10); otherwise, the search direction d^k is generated by the projected gradient method, which is the first formula of (10). Furthermore, combining with Armijio-type line search, it can be proved that the proposed PASHS algorithm guarantees the feasibility and global convergence for non-linear optimization problem (7).

3.2. Convergence Analysis

In this subsection, to further investigate the convergence of the Algorithm 1 (PASHS) for non-linear optimization problem (7), some basic assumptions should be revisited and introduced in this subsection.

Assumption 1. The level set

$$D = \{x \in \Omega | \Gamma(x) \leq \Gamma(x^0), \Lambda x = b\} \tag{17}$$

is bounded.

Assumption 2. Given that the objective function $\Gamma : R^n \to R$ is continuously differentiable on an open set $N \subseteq D$ and its gradient is Lipschitz continuous, there exists a positive constant $W > 0$ that satisfies the following inequality:

$$\|\nabla\Gamma(x) - \nabla\Gamma(y)\| \leq W\|x - y\|, \ \forall x, y \in N. \tag{18}$$

As $\{\Gamma(x^k)\}$ is a descending sequence, it is clear that the sequence $\{x^k\}$ generated by Algorithm 1 (PASHS) is contained in D. In addition, according to Assumption 1, it is inferred that the gradient of Γ is bounded, i.e., there exists a positive constant $\gamma > 0$ such that

$$\|\nabla\Gamma(x)\| \leq \gamma, \ \forall x \in D. \tag{19}$$

Since the matrix $\mathscr{P}$ is a projected matrix, suppose that there exists a positive constant $C > 0$, and the following inequality can be obtained:

$$\|\mathscr{P}\nabla\Gamma(x)\| \leq C, \ \forall x \in D. \tag{20}$$

Lemma 2. Assume that the iterative sequence $\{x^k\}$ generated by Algorithm 1 (PASHS). The step size η^k is obtained via the Armijo line search rule (16), and then there exists a positive constant $c_0 > 0$ such that the following inequality holds

$$\eta_k \geq c_0 \frac{\|\mathscr{P}^k\nabla\Gamma^k\|^2}{\|d^k\|^2} \tag{21}$$

for sufficiently large k.

Proof. According to Armijio-type linear search rule (16), the following inequality can be obtained:

$$\sum_{i=1}^{\infty} -\delta\eta_k(\mathscr{P}^k\nabla\Gamma^k)^T d^k \leq \Gamma(x^0) - \Gamma(x^*) < +\infty. \tag{22}$$

Combined (11) and (12), and the properties of the projected matrix $\mathscr{P}^k$, the inequality can be generalized as follows

$$\sum_{i=1}^{\infty} \eta_k\|\mathscr{P}^k\nabla\Gamma^k\|^2 = \sum_{i=1}^{\infty} \eta_k(\nabla\Gamma^k)^T\mathscr{P}^k\nabla\Gamma^k$$
$$= -\sum_{i=1}^{\infty} \eta_k(\mathscr{P}^k\nabla\Gamma^k)^T d^k < +\infty. \tag{23}$$

Now, the following two cases can be taken into account and be utilized to prove (21).

Case 1: If $\eta_k = 1$, according to Equations (11), (12), and (23), and by applying the Cauchy-Schwarz inequality, we can derive that

$$\|\mathscr{P}^k\nabla\Gamma^k\|^2 = |(\mathscr{P}^k\nabla\Gamma^k)^T d^k| \leq \|\mathscr{P}^k\nabla\Gamma^k\| \cdot \|d^k\|.$$

Thus, the inequality (21) holds.

Case 2: If $\eta_k < 1$, assume that the Armijio-type line search rule is not true, there thus exists a positive constant $\rho^{-1}\eta_k$ such that the following inequality holds true:

$$\Gamma(x^k + \rho^{-1}\eta_k d^k) - \Gamma(x^k) > \delta\rho^{-1}\eta_k(\mathscr{P}^k\nabla\Gamma^k)^T d^k. \tag{24}$$

Using the mean-value theorem and Assumption 1, there exists a positive constant $\xi_k \in (0, 1)$ such that $x^k + \xi_k\rho^{-1}\eta_k d^k \in D$ and

$$\Gamma(x^k + \rho^{-1}\eta_k d^k) - \Gamma(x^k) = \rho^{-1}\eta_k\nabla\Gamma(x^k + \xi_k\rho^{-1}\eta_k d^k)^T d^k$$
$$= \rho^{-1}\eta_k(\nabla\Gamma^k)^T d^k + \rho^{-1}\eta_k(\nabla\Gamma(x^k + \xi_k\rho^{-1}\eta_k d^k) - \nabla\Gamma^k)^T d^k$$
$$\leq \rho^{-1}\eta_k(\nabla\Gamma^k)^T d^k + W\rho^{-2}(\eta_k)^2\|d^k\|^2. \tag{25}$$

Combining inequality (24) and Theorem 1, the following inequality can be directly computed as

$$\eta_k \geq \frac{(1 - \delta)\rho}{W} \frac{\|\mathscr{P}^k\nabla\Gamma^k\|^2}{\|d^k\|^2}. \tag{26}$$

Let $c_0 = \min\{1, (1 - \delta)\rho/W\}$, the conclusion is true. $\square$

Lemma 3. Suppose that Assumption 2 holds. The iterative sequence $\{x^k\}$ is generated by Algorithm 1 (PASHS), and then the search direction d^k defined by (10) is bounded, in other words, there exists a positive constant $M \geq 0$ such that

$$\|d^k\| \leq M, \forall k \in \mathbb{N}^*. \tag{27}$$

Proof. According to the Assumption 2 and Algorithm 1 (PASHS), the following inequality can be directly obtained:

$$(d^{k-1})^T v^{k-1} > \gamma_0\eta_{k-1}\|d^{k-1}\|^2.$$

In addition, in term of the search direction (10) and Theorem 1, the following inequality can be derived as

$$
\begin{aligned}
\|d^k\| \leq &\|\mathscr{P}^k \nabla \Gamma^k\| + |\beta_k^{HS}| \cdot \|d^{k-1}\| + |\zeta_k| \cdot \|\mathscr{P}^k z^{k-1}\| \\
\leq &\|\mathscr{P}\nabla\Gamma^k\| + \frac{\|(\mathscr{P}^k\nabla\Gamma^k)^T z^{k-1}\|}{\gamma_0 \eta_{k-1}\|d^{k-1}\|^2}\|d^{k-1}\| \\
&+ \frac{\|(\mathscr{P}^k\nabla\Gamma^k)^T d^{k-1}\|}{\gamma_0 \eta_{k-1}\|d^{k-1}\|^2}\|\mathscr{P}^k z^{k-1}\| \\
\leq &\|\mathscr{P}^k\nabla\Gamma^k\| + \frac{\|(\mathscr{P}^k\nabla\Gamma^k)\|}{\gamma_0\eta_{k-1}\|d^{k-1}\|}(\|z^{k-1}\| + \|\mathscr{P}^k z^{k-1}\|) \\
\leq &\|\mathscr{P}^k\nabla\Gamma^k\| + \frac{\|(\mathscr{P}^k\nabla\Gamma^k)\|W(1+\lambda_{\max}(\mathscr{P}^k))}{\gamma_0} \\
\leq &C(1 + \frac{W(1+\lambda_{\max}(\mathscr{P}^k))}{\gamma_0}).
\end{aligned}
$$
(28)

where $\lambda_{\max}(\mathscr{P}^k)$ is the maximum eigenvalue of the projected matrix $\mathscr{P}^k$. $\qquad\square$

Theorem 2. Suppose that Assumption 1 holds. The iterative sequence $\{x^k\}$ is generated by Algorithm 1 (PASHS), and then

$$
\lim_{k\to\infty}\inf\|\mathscr{P}^k\nabla\Gamma^k\| = 0.
$$
(29)

Proof. According to (21), there exists a positive constant c_0 such that

$$
\eta_k\|d^k\|^2 \geq c_0\|\mathscr{P}^k\nabla\Gamma^k\|^2.
$$
(30)

Combining Algorithm 1 with Lemma 3, it implies that

$$
\lim_{k\to\infty}\eta_k\|d^k\|^2 = 0.
$$

The following inequality can be generalized as $k \to \infty$,

$$
0 = \lim_{k\to\infty}\frac{\eta_k}{c_0}\|d^k\|^2 \geq \lim_{k\to\infty}\inf\|\mathscr{P}^k\nabla\Gamma_k\|^2 \geq 0.
$$
(31)

Hence $\lim_{k\to\infty}\inf\|\mathscr{P}^k\nabla\Gamma^k\| = 0$. $\qquad\square$

Remark. Owing to Theorem 2, it can be seen that the Algorithm 1 (PASHS) is globally convergent for the non-linear optimization problem (7). Combing the ESN learning algorithm and Algorithm 1 (PASHS), the optimal controller of the MPC problem can thus be solved rapidly; this is used for the patients through a lower-limb rehabilitation robot with passive and active rehabilitation training.

4. SIMULATIONS AND RESULTS

In this section, the proposed PASHS algorithm with MPC technique is applied to the passive rehabilitation training of the two-link lower-limb rehabilitation robot. Moreover, combining the ESN model and intention recognition, the MPC and PASHS algorithm also are utilized to active rehabilitation training.
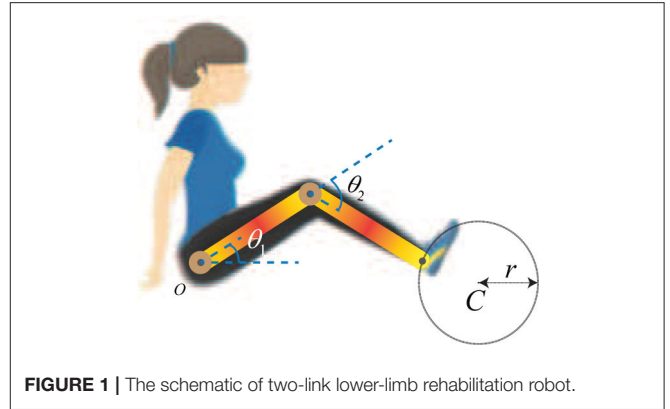
FIGURE 1 | The schematic of two-link lower-limb rehabilitation robot.

4.1. Two-Link Lower-Limb Rehabilitation Robot With MPC

The general dynamic model of two-link lower-limb rehabilitation robot is shown as follows (He et al., 2015):

$$
\mathbf{D}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) = \boldsymbol{\tau},
$$
(32)

where $\mathbf{q}$, $\dot{\mathbf{q}}$, $\ddot{\mathbf{q}} \in R^2$ are angle, angular velocity and angular acceleration of hip and knee, respectively; $\boldsymbol{\tau} \in R^2$ is a torque for the rehabilitation robot, which represents admissible control inputs; $\mathbf{D}(\mathbf{q}) \in R^{2\times2}$ is a positive-definite inertia matrix; $\mathbf{C}(\mathbf{q},\dot{\mathbf{q}}) \in R^{2\times2}$ is a centrifugal and Coriolis term; and $\mathbf{G}(\mathbf{q}) \in R^2$ is related to gravity term. The state space expression of (32) can be described as

$$
\begin{cases}
\begin{bmatrix}\dot{\mathbf{q}}\\\ddot{\mathbf{q}}\end{bmatrix} = \begin{bmatrix}\mathbf{0} & I_{2\times2}\\\mathbf{0} & -\mathbf{D}^{-1}(\mathbf{q})\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\end{bmatrix}\begin{bmatrix}\mathbf{q}\\\dot{\mathbf{q}}\end{bmatrix} + \begin{bmatrix}\mathbf{0}\\\mathbf{D}^{-1}(\mathbf{q})\end{bmatrix}\boldsymbol{\tau}\\
\qquad - \begin{bmatrix}\mathbf{0}\\\mathbf{D}^{-1}(\mathbf{q})\mathbf{G}(\mathbf{q})\end{bmatrix}, \\
\mathbf{y} = \mathbf{h}(\mathbf{q}),
\end{cases}
$$
(33)

where $\mathbf{y} \in R^2$ is the end-effector position coordinates, $\mathbf{h}(\cdot)$ is a function mapping angles of the rehabilitation robot to the position coordinates. The schematic of a two-link lower-limb rehabilitation robot is shown in **Figure 1**.

As shown in **Figure 3**, $\theta_1 = q_1$, $\theta_2 = q_2$, C and r represent the the hip joint angle, the knee joint angle, center, and radius of the reference trajectory (which can be defined as a circle), respectively. The lengths of the links are $l_1 = 0.35$ m and $l_2 = 0.32$ m; the mass and inertia of two links are $m_1 = 1.8$ kg, $m_2 = 1.65$ kg and $I_1 = \frac{1}{4}m_1 l_1^2$ kg·m, $I_2 = \frac{1}{4}m_2 l_2^2$ kg·m, respectively; the gravity constant is $g = 9.801$ m/s^2. In addition, the center and radius are $C = (0.5, 0)$ and $r = 0.1$ m, respectively.

The parameters of the algorithm 1 (PASHS) are chosen as follows:

$$
\varepsilon = 10^{-6},\ \psi_0 = 10^{-5},\ \gamma_0 = 10^{-3},\ \delta = 0.0001,\ \rho = 0.5,\ M^0 = \Lambda,
$$

and the initial step size is selected as (Dai, 2011)

$$
\eta_0 = \left|\frac{-\gamma_0 \nabla\Gamma^{kT}d^k}{d^{kT}(\nabla\Gamma(x^k + \gamma_0 d^k) - \nabla\Gamma^k)}\right|.
$$

FIGURE 2 | The simulation results with passive rehabilitation training, **(A)** the angles of lower-limb rehabilitation robot with constraints (34), and **(B)** the angular velocities of lower-limb rehabilitation robot with constraints (34).

FIGURE 3 | The simulation results with passive rehabilitation training, **(A)** the phase portraits for angles of lower-limb rehabilitation robot with constraints (34), **(B)** the phase portraits for angular velocities of lower-limb rehabilitation robot with constraints (34).

Due to the proposed MPC model in section 3, the parameters of the MPC will be defined as follows:

$$Q = R = I_{2 \times 2},$$

the prediction horizon is $N = 5$ and the control horizon is $N_u = 5$; the sampling time is $h = 0.01$s. The initial state of end-effector position coordinates is $y_0 = (0.6, 0)$, and the motion-task duration is 25 s with 5 s per cycle. The following experiments are conducted under different constraints of torque: while the hip joint angle and the angular velocity are constrained in interval $\left[-\frac{2}{3}\pi, \frac{2}{3}\pi\right]$, and $[-\pi, \pi]$, the knee joint angle and the angular velocity are constrained in interval $\left[-\frac{4}{3}\pi, 0\right]$, and $[-\pi, \pi]$ (Jin and Zhang, 2011).

4.2. The Passive Rehabilitation Training With Different Torques Constraints

Example 1: Consider the following control torques constraints

$$-15 \text{ N·m} \leq \tau_1, \tau_2 \leq 15 \text{ N·m}, \qquad (34)$$

where τ_1 and τ_2 correspond to the torques of the hip joint and the knee joint of a two-link lower-limb rehabilitation robot,

respectively. The numerical results of this situation are shown in **Figures 2**–**4**.

Figure 2 represents the curves of angle and angular velocities of the hip and knee for the two-link lower-limb rehabilitation robot, and **Figure 3** is the limit cycles of angle and angular velocities. From **Figures 2**, **3**, it is further inferred that the angles and angular velocities of the lower-limb rehabilitation robot present the periodic properties; it also verifies that the proposed approach is feasible and effective. **Figure 4A** denotes the control torque vs. time, and the hip joint and knee joint of the rehabilitation robot can be controlled by torques τ_1 and τ_2, respectively. As shown in **Figure 4A**, it can be seen that the control torques change periodically for two-link lower-limb rehabilitation robot, which can help the injured patients to do rehabilitation training stably via Algorithm 1 (PASHS) with MPC technique. **Figure 4B** represents the tracking errors of the real position of end-effector and desired trajectory, while e_x and e_y are the tracking errors of horizontal ordinate and longitudinal coordinates. As shown in **Figure 4B**, the absolute value of the tracking errors is also smaller than 0.002 m, which also infers that the lower-limb rehabilitation robot could implement the passive rehabilitation training efficiently by the desired trajectory and MPC technique. It thus further
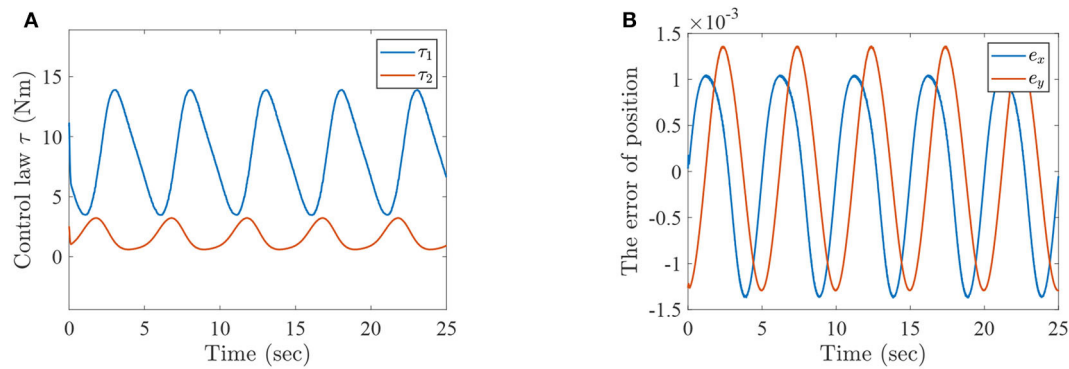
FIGURE 4 | The simulation results with passive rehabilitation training, **(A)** the torques of lower-limb rehabilitation robot with constraints (34), and **(B)** the tracking errors of lower-limb rehabilitation robot with constraints (34).
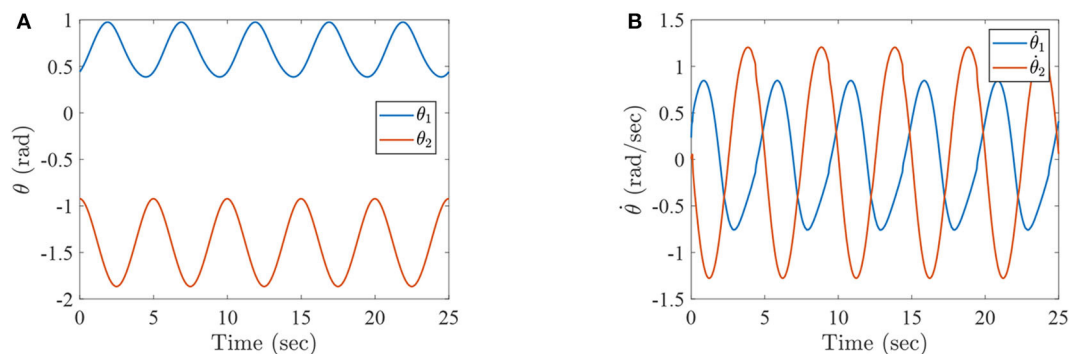
FIGURE 5 | The simulation results with passive rehabilitation training, **(A)** the angles of lower-limb rehabilitation robot with constraints (35), and **(B)** the angular velocities of lower-limb rehabilitation robot with constraints (35).

demonstrates that the theoretical analyses are feasible and reliable. Besides, it is very important to take into consideration the energy consumption of rehabilitation training in real-world rehabilitation implementations, therefore, control torques should be constrained within relatively reasonable bounds. As shown in **Figure 4A**, the optimal control input is obtained by online solving of the MPC problem via Algorithm 1 (PASHS). However, it can be seen from **Figure 4A**, that the bounded condition is too large for non-linear optimization problem with MPC model, that is, the real-time control torque τ_1 is around 13 N·m. In other words, to achieve the low-energy consumption, the boundary constraints can be reduced to around 10 N·m, which are utilized to control the two-link lower limb rehabilitation robot to realize the rehabilitation cycle movement of the injured lower limb.

Example 2: Consider the following control torques with constraint conditions

$$-10\,\mathrm{N\cdot m} \leq \tau_1, \tau_2 \leq 10\,\mathrm{N\cdot m}. \tag{35}$$

The numerical results of this situation are shown in **Figures 5**–**7**.

Figure 5 represents the curves of angle and angular velocities for the two-link lower-limb rehabilitation robot, and **Figure 6** shows the limit cycles of angle and angular velocities, respectively. As can be seen from **Figures 5**, **6**, the angular velocities of the

lower limb rehabilitation robot are affected by the control torques with constraint conditions limited to −10–10 N·m. However, the injured limb can stably complete rehabilitation training activities via the algorithm 1 (PASHS) with MPC technique. **Figure 7A** shows the control torque vs. time, and it can be seen that the control input τ_1 could be constrained in between −10 and 10 N·m, and the smoothness of angular velocities maybe influenced by the constraint conditions. However, it further infers that the stability can be maintained for the two-link lower limb rehabilitation robot. **Figure 7B** plots the tracking errors of the real trajectories of end-effector and desired trajectories, while e_x and e_y are the same with the definition of Example 1. The proposed approach is therefore suitable for passive rehabilitation training of lower-limb rehabilitation robot.

Example 3: This example shows a comparison of different algorithms with the MPC solution.

In order to compare the advantages of PASHS algorithm, sequential quadratic programming (SQP) is selected to compare with our algorithm (Sun et al., 2020). The simulation problem is chosen as example 1, and the results are as follows:

Figure 8A represents the horizontal ordinate tracking errors e_x of lower-limb rehabilitation robots with passive rehabilitation training, and **Figure 8B** means the longitudinal ordinate tracking errors e_y. From those two figures, it can be seen that the
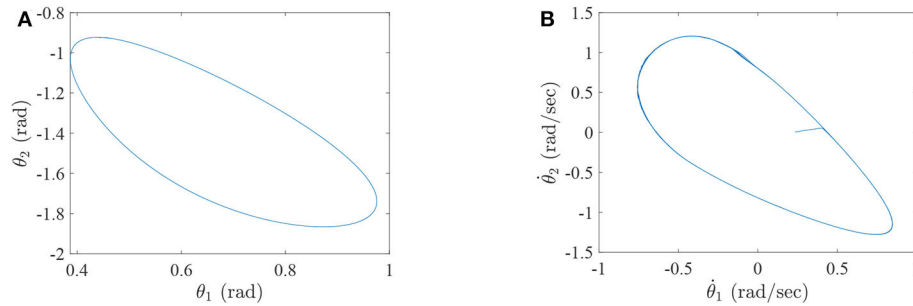
FIGURE 6 | The simulation results with passive rehabilitation training, **(A)** the phase portraits for angles of lower-limb rehabilitation robot with constraints (35), and **(B)** the phase portraits for angular velocities of lower-limb rehabilitation robot with constraints (35).
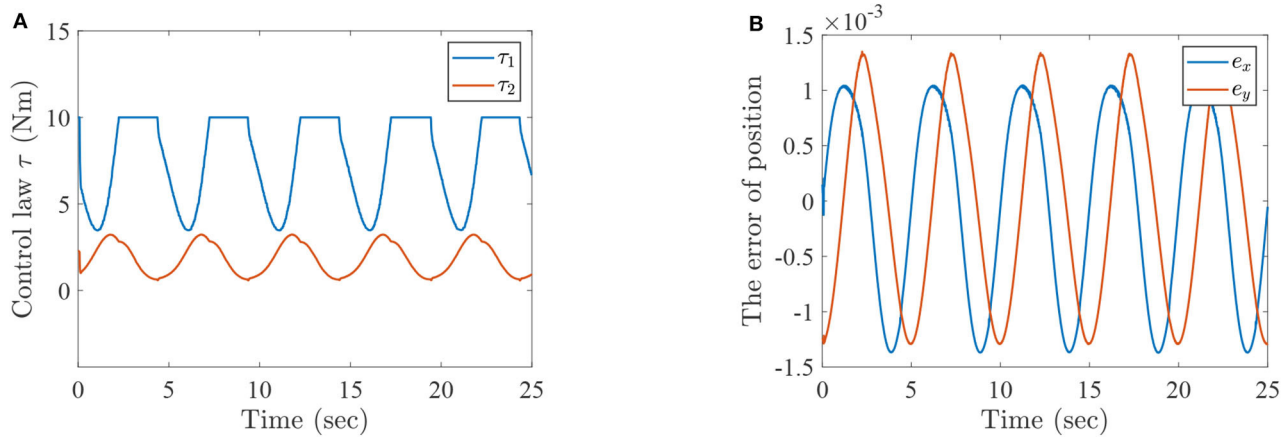
FIGURE 7 | The simulation results with passive rehabilitation training, **(A)** the torques of lower-limb rehabilitation robot with constraints (35), and **(B)** the tracking errors of lower-limb rehabilitation robot with constraints (35).
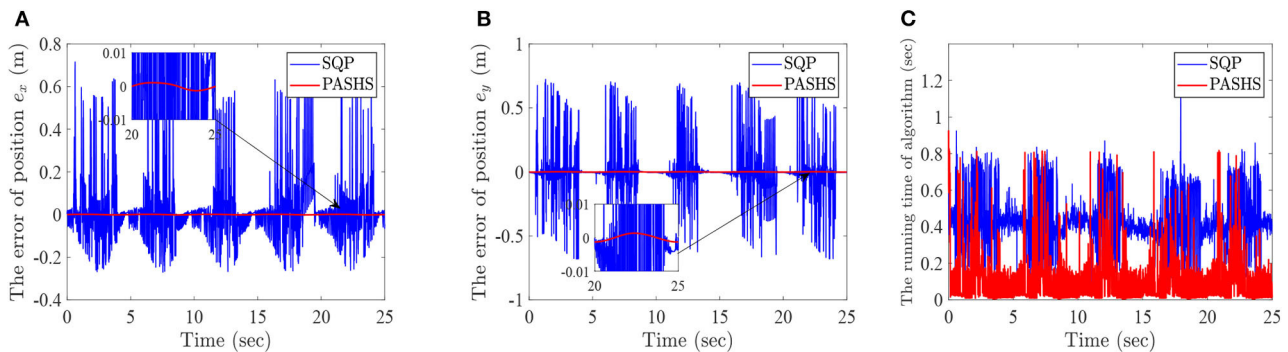
FIGURE 8 | The simulation results with PASHS and SQP, **(A)** the tracking errors of horizontal ordinate, **(B)** the tracking errors of longitudinal coordinates, and **(C)** the running time of algorithm at every time.

tracking errors of SQP are almost > 0.1 m at every iteration, and sometimes the errors were closed to 1m. This is because an optimal solution for SQP may not be in a feasible region. However, the tracking errors of PASHS are always smaller than 0.01 at every time. The optimal solution of the PASHS

algorithm was satisfied by the constraint conditions because of the projective matrix and active set. PASHS was therefore more suitable for the MPC of lower-limb rehabilitation robots. **Figure 8C** is the running time of two algorithms at every optimization; according to this figure, the running time of PASHS

FIGURE 9 | The simulation results with 1.5 times parameters' perturbation, **(A)** the angles of lower-limb rehabilitation robot with constraints (34), and **(B)** the angular velocities of lower-limb rehabilitation robot with constraints (34).

FIGURE 10 | The simulation results with 1.5 times parameters' perturbation, **(A)** the torques of lower-limb rehabilitation robot with constraints (34), and **(B)** the tracking errors of lower-limb rehabilitation robot with constraints (34).

was nearly always smaller than 0.2 s, and the SQP running time was around of 0.4 s. This can account for the real-time computing capability of PASHS algorithm with the MPC technique.

Example 4: The example with parameters' perturbation 1.5 times.

This example is reported the influence of uncertainties in system, and the 1.5 times parameters' perturbation is introduced into lower-limb rehabilitation robots. During the simulation, the parameters are selected as $l_1 = 0.35$ m, $l_2 = 0.32$ m, $m_1 = 2.7$ kg, $m_2 = 2.475$ kg, $I_1 = 0.0827$ kg·m, and $I_2 = 0.0634$ kg·m. Other conditions are the same as Example 1. The results of this example are shown as follows.

Figures 9A,B represent the angles and angular velocities of lower-limb rehabilitation robot with 1.5 times parameters' perturbation, respectively. As can be seen from these figures, the angles and angular velocities are changed periodically and stably, although the model is disturbed. **Figure 10A** is the torques of lower-limb rehabilitation robot which is subjected to 1.5 times parameter perturbation, and **Figure 10B** represents the horizontal ordinate tracking errors e_x and the longitudinal ordinate tracking errors e_y of lower-limb rehabilitation robots with 1.5 times parameters' perturbation. From **Figure 10A**, we

can find that torques are limited between -15 and 15 N·m because the mass of lower-limb rehabilitation robot is added. However, the absolute value of tracking errors are smaller than 0.0015 m according to **Figure 10B**. The PASHS algorithm could thus solve MPC problems of the lower-limb rehabilitation robot with uncertainties in the model, and high accuracy could also be guaranteed.

4.3. sEMG-Based Active Rehabilitation Training

In this subsection, the active intention of injured patients is regarded as one of the most important rehabilitation steps. Furthermore, the joint trajectories of injured lower limb can be identified via the mentioned ESN model based on the active motion intention, which can be seen as the desired trajectories of lower limb rehabilitation robots. A numerical simulation is illustrated and analyzed for two-link lower limb rehabilitation robot with ESN model and MPC technique. The technical diagram of sEMG-based active rehabilitation training and intention recognition is shown in **Figure 11**.

The active rehabilitation training consists of two parts. The first one is intention recognition, which collects and preprocesses
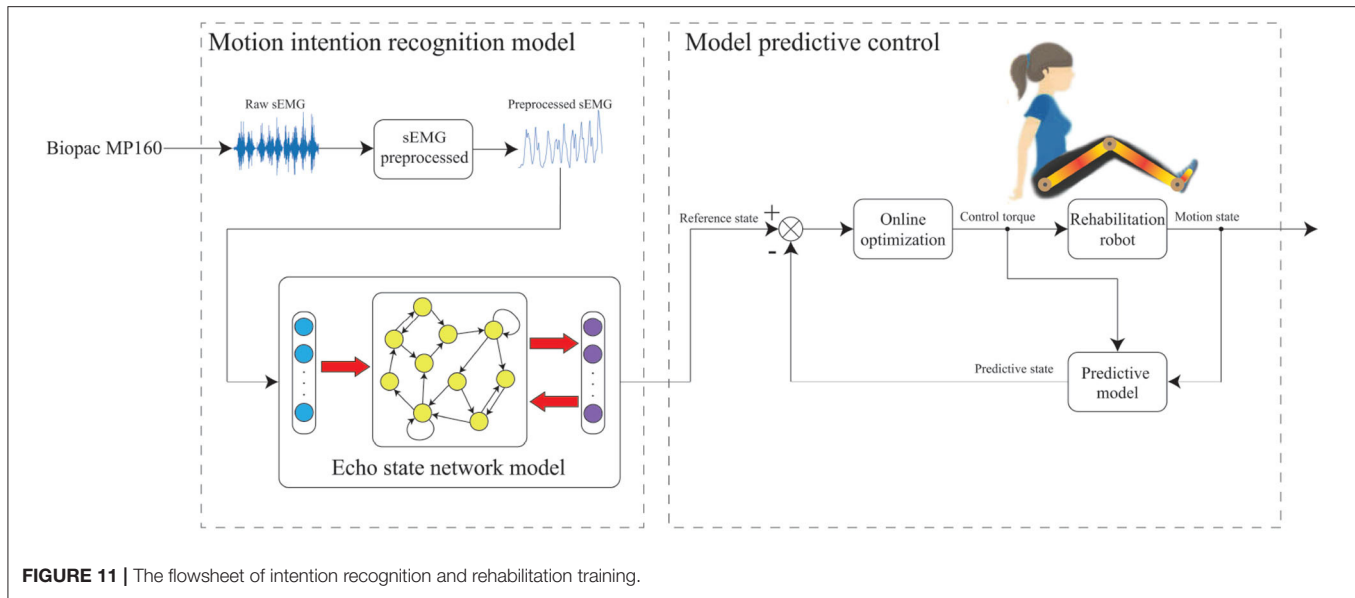
FIGURE 11 | The flowsheet of intention recognition and rehabilitation training.

raw sEMG signals, and then motion intention is identified by the ESN model. The details of first part is described in the following description.

During the data acquisition stage, a subject sits on a chair and swings the shank periodically. The sEMG signals of seven muscles of leg, which include the vastus rectus muscle (VR), semitendinosus muscle (SM), tibialis anterior muscle (TA), gastrocnemius muscle (GM), vastus lateralis muscle (VL), biceps muscle of thigh (BM), and extensor pollicis longus (EP), need to be recorded through data acquisition unit, respectively (Tong et al., 2014). The acquisition device is BIOPAC MP160, which can simultaneously capture eight channels of sEMG signals at the default 2 kHz sample rate. Angles and angular velocities of knee and ankle are recorded by inertial measurement unit (IMU), which selects 100 Hz as the sample rate. Due to the sample rate of sEMG signals is higher than IMU, the sub-sampling technology should be implemented by (Zhang et al., 2012)

$$\mathrm{sEMG_{pre}}(k) = \mathrm{sEMG}(20k - 19),$$

where $\mathrm{sEMG_{pre}}(k)$ represents the sEMG signals after sub-sampling at k times. The position of the electrodes and raw sEMG signals are shown in **Figure 12**.

We then use neural network technology to establish the relationship between sEMG signals and motion state. This is due to the original sEMG signals being contaminated by different measurement noises, such as direct current bias and baseline noise (Law et al., 2011). The raw sEMG signals need to be preprocessed, which includes a high-pass filter with 50 Hz high cut-off frequency, full-wave rectification technology, low-pass filter with 5 Hz low cut-off frequency and normalized technology (Han et al., 2015). The sEMG signals can be seen as the input signals of neural network when the noise of raw sEMG signals is eliminated by the mentioned methods.

The ESN model is a kind of recurrent neural network that is composed of an input layer, a hidden layer where neurons

interconnect randomly, and an output layer. The network architecture is depicted in **Figure 13**.

The mathematical model of the ESN model can be obtained as

$$\begin{aligned}
\mathcal{X}^{k+1} &= \mathcal{F}(\mathfrak{M}_{\mathcal{X}}\mathcal{X}^k + \mathfrak{M}_{\mathcal{U}}\mathcal{U}^{k+1} + \mathfrak{M}_{\mathcal{F}}\mathcal{Y}^k), \\
\mathcal{Y}^{k+1} &= \mathfrak{M}_{\mathcal{Y}}\mathcal{X}^{k+1},
\end{aligned} \tag{36}$$

where $\mathcal{F}(\cdot)$ is an activation function and commonly generates from $\mathcal{F}(\mathrm{x}) = \tanh(\mathrm{x})$; $\mathfrak{M}_{\mathcal{X}} \in R^{l\times l}$, $\mathfrak{M}_{\mathcal{U}} \in R^{l\times n}$, $\mathfrak{M}_{\mathcal{F}} \in R^{l\times m}$, $\mathfrak{M}_{\mathcal{Y}} \in R^{m\times l}$ are the internal connection weight of the hidden layer, the input layer to the hidden layer connection weight matrix, the output layer to the hidden layer feedback weight matrix, and the hidden layer to the output layer connection weight matrix; $\mathcal{X}$ and $\mathcal{Y}$ are the echo state and output vectors of the ESN model, respectively. Assume that $\mathfrak{M}_{\mathcal{X}}$, $\mathfrak{M}_{\mathcal{U}}$, and $\mathfrak{M}_{\mathcal{F}}$ are unmodifiable during the ESN model training (Pan and Wang, 2012). The ESN algorithm with off-line learning is summarized as follows.

Algorithm 2. (ESN learning algorithm)

Step 0. Initialize echo state $\mathcal{X}^0$ and randomly obtain a matrix $\mathfrak{M}$. Normalize $\hat{\mathfrak{M}} = \mathfrak{M}/|\lambda_{\max}|$ and generate $\mathfrak{M}_{\mathcal{X}} = \alpha_{\mathcal{X}}\hat{\mathfrak{M}}$, $\mathfrak{M}_{\mathcal{U}} \in R^{l\times n}, \mathfrak{M}_{\mathcal{F}} \in R^{l\times m}$, where $\alpha_{\mathcal{X}} < 1$ is a spectral radiuses of $\mathfrak{M}_{\mathcal{X}}$ and $|\lambda_{\max}|$ is a spectral radius of $\mathfrak{M}$.

Step 1. Compute the echo state by

$$\mathcal{X}^{k+1} = \mathcal{F}(\mathfrak{M}_{\mathcal{X}}\mathcal{X}^k + \mathfrak{M}_{\mathcal{U}}\hat{\mathcal{U}}^{k+1} + \mathfrak{M}_{\mathcal{F}}\hat{\mathcal{Y}}^k)$$

for $k = 0, 1, \ldots, N$, where $\hat{\mathcal{U}}^k$ and $\hat{\mathcal{Y}}^k$ are the kth input and output reference data from the training dataset.

Step 2. Collect the reservoir state **X** and target state **Y** as follows:

$$\mathbf{X} = [\mathcal{X}^1, \ldots, \mathcal{X}^N], \mathbf{Y} = [\hat{\mathcal{Y}}^1, \ldots, \hat{\mathcal{Y}}^N].$$

Step 3. Off-line compute matrix $\mathfrak{M}_{\mathcal{Y}} = (\mathbf{X}^+\mathbf{Y})^T$, where $\mathbf{X}^+$ is the pseudo-inverse of **X**.
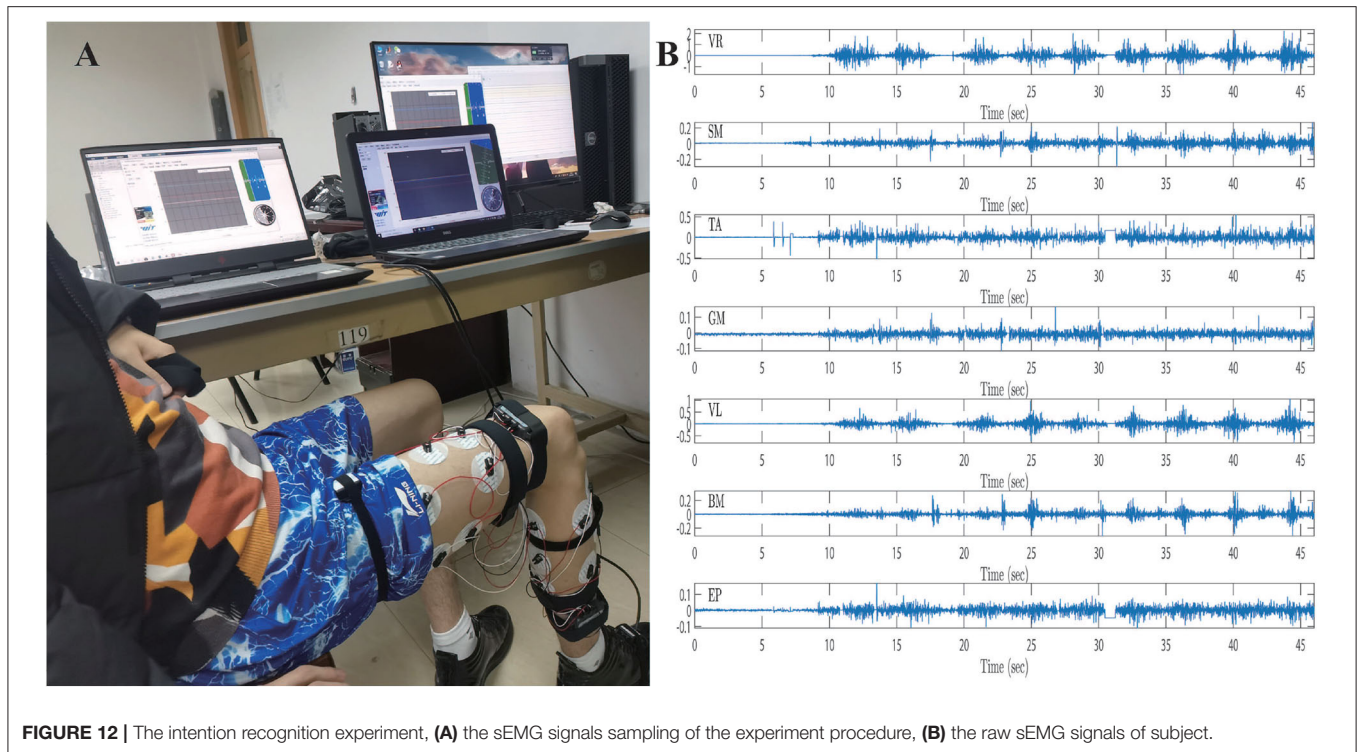
FIGURE 12 | The intention recognition experiment, **(A)** the sEMG signals sampling of the experiment procedure, **(B)** the raw sEMG signals of subject.
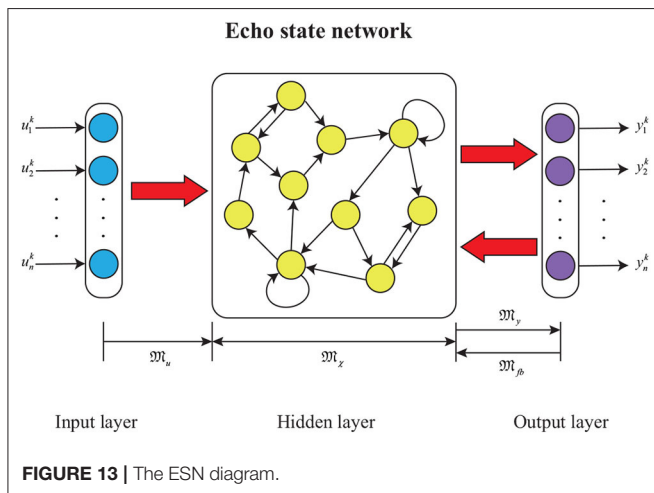
FIGURE 13 | The ESN diagram.

During the training process, the preprocessed sEMG signals are recorded by the Biopac MP160 system with seven channels from 0 to 46 s. Furthermore, the joint trajectories of injured lower limb are recorded by IMU, the data set from 0 to 32 s is then collected as a training set, and the remainder of the data set is regarded as a test set. The training set is utilized to train the ESN model, and the testing set is exploited to simulate lower limb rehabilitation robots with active rehabilitation training. The real angles and angular velocities are recorded by IMU, which aims at demonstrating the accuracy of proposed method. The ESN model has seven input neurons and four output neurons, the numbers of ESN hidden layer neurons are 100, and the parameters are

selected as follows: $\alpha_{\mathcal{X}} = 0.5$. The results of ESN training and testing are shown in **Figures 14**, **15**.

Figure 14 represents the intention recognition results of injured lower limb via ESN model, where θ_1 and θ_2 mean knee and ankle angles, respectively. Red solid lines denote real joint trajectories of knee and ankle angles, and blue solid lines represent training and testing results of ESN learning algorithm. **Figure 15** shows the training and testing results of knee and ankle angular velocities through ESN learning algorithm. $\dot{\theta}_1$ and $\dot{\theta}_2$ mean the angular velocities for knee and ankle, which are shown by blue solid lines. Red solid lines represent real angular velocities, which are recorded by IMU. As can be seen from **Figure 14**, the injured lower limb swings the calf at the 5th second, and then the knee periodically stretches and flexes about at about 41 s. When the knee joint angle reaches 0 rad, the knee joint swings to its maximum position. During the swing phase, the knee joint can flex more than −1.8 rad. The angle of the ankle joint is between 1.8 and 2.2 rad, which is always plantar flexion. **Figure 15** shows that the angular velocity of the knee joint alternates between −2 and 2 rad/s with a cycle of about 4 s, while the angular velocity of the ankle joint varies slightly between −0.5 and 0.5 rad/s. The motion intention of injured lower limbs can be identified by the ESN learning algorithm with multichannel sEMG signals from 32 to 45.3 s. Meanwhile, it is also inferred that the proposed method shows superior performance for intention identification of injured lower limb.

The second one is an MPC problem; the joint trajectories of injured lower limb can be identified via an ESN model based on active motion intention, which is viewed as desired trajectories of the two-link lower limb rehabilitation robot. The control law
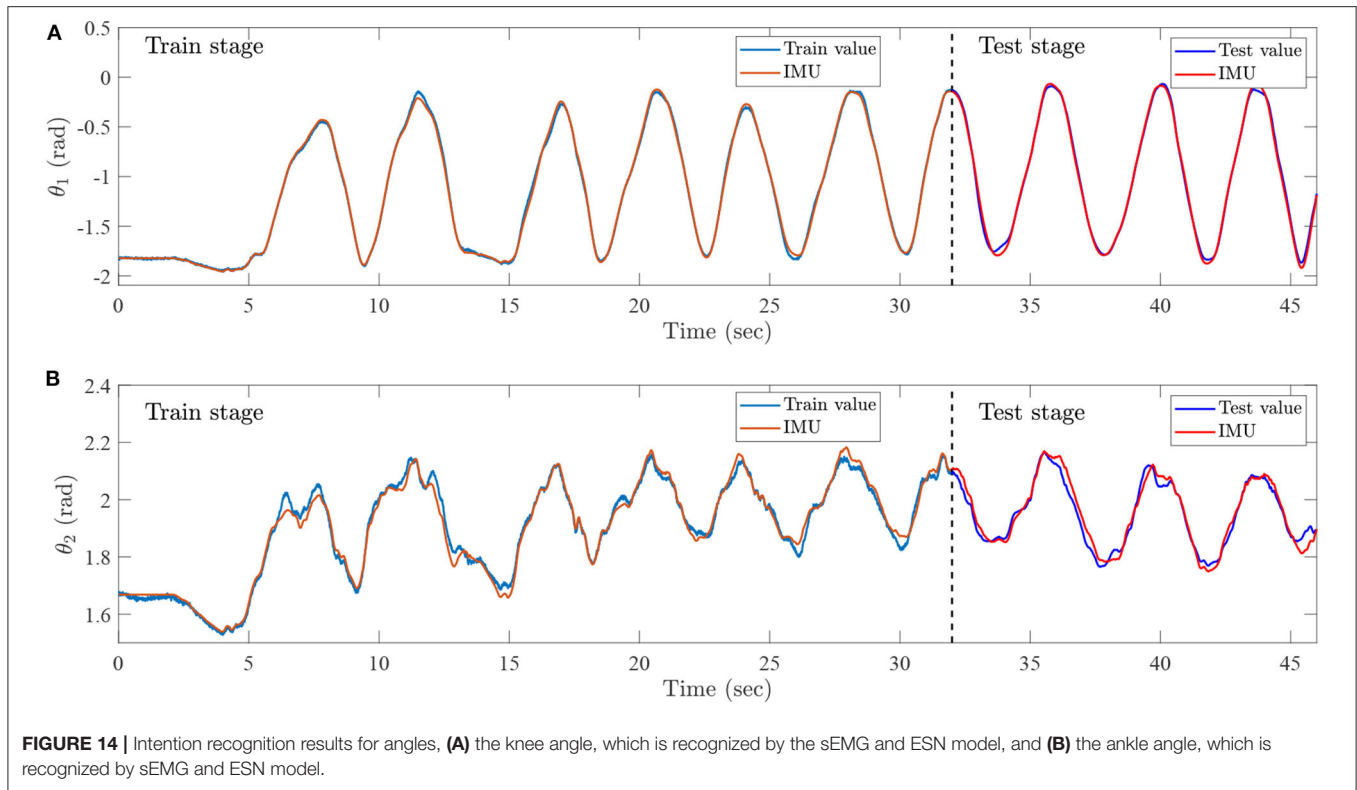
FIGURE 14 | Intention recognition results for angles, **(A)** the knee angle, which is recognized by the sEMG and ESN model, and **(B)** the ankle angle, which is recognized by sEMG and ESN model.
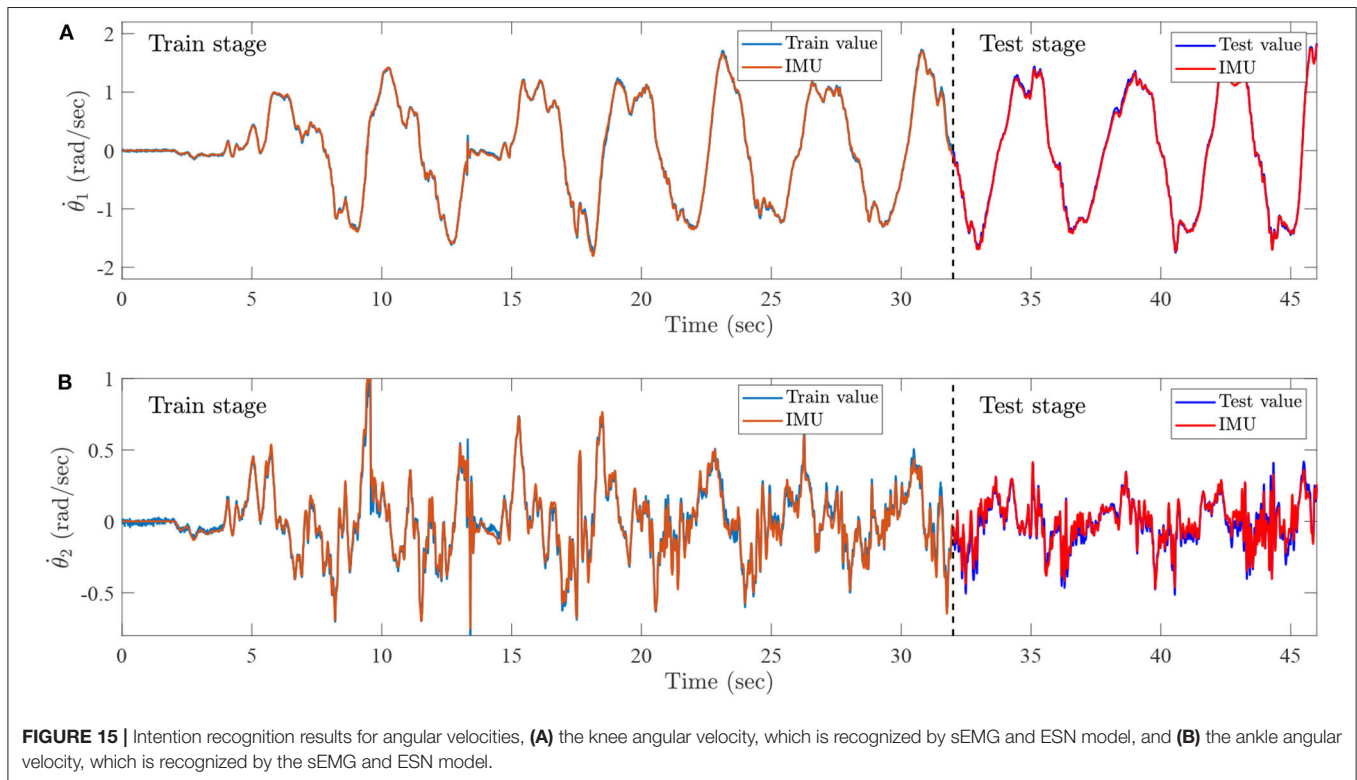
FIGURE 15 | Intention recognition results for angular velocities, **(A)** the knee angular velocity, which is recognized by sEMG and ESN model, and **(B)** the ankle angular velocity, which is recognized by the sEMG and ESN model.

generated by the Algorithm 1 (PASHS) is transmitted to two-link lower limb rehabilitation robot, which aims at assisting patient to do rehabilitation training. The next predictive state also can be computed by the optimization results of MPC problem (2), which feeds back to the two-link lower limb rehabilitation robot system. Meanwhile, the MPC problem can be seen as follows:

FIGURE 16 | The simulation results with active rehabilitation training, **(A)** the angles of lower-limb rehabilitation robot based on active intentions, **(B)** the angular velocities of lower-limb rehabilitation robot based on active intentions, and **(C)** the torques of lower-limb rehabilitation robot.
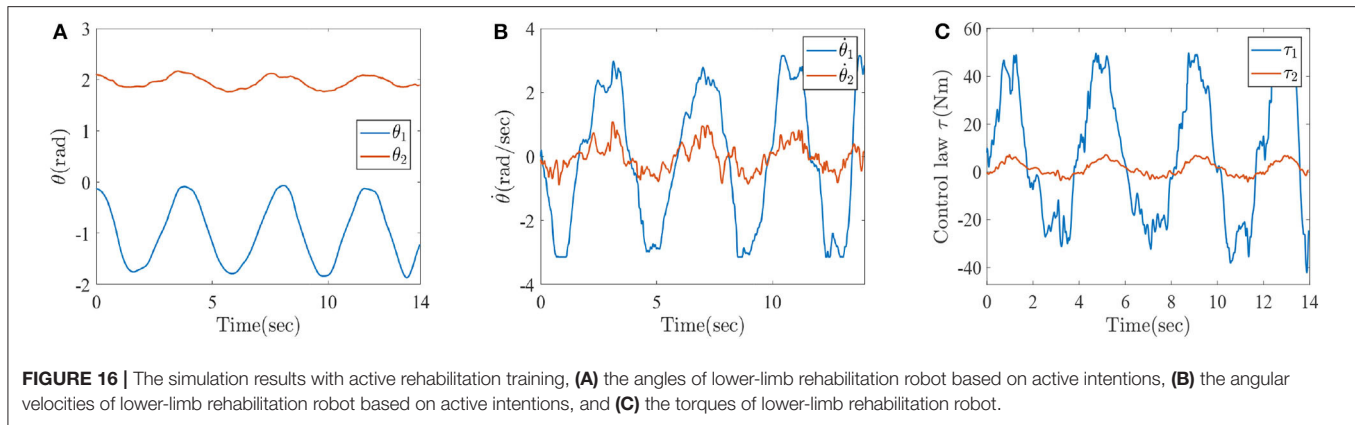
$$\min_{\boldsymbol{\theta}^k, \dot{\boldsymbol{\theta}}^k, \boldsymbol{\tau}^k} \sum_{i=1}^{N} \left\| \boldsymbol{\theta}(k+i|k) - \boldsymbol{\theta}^{\mathrm{d}}(k+i|k) \right\|_Q^2 + \sum_{j=0}^{N_\tau - 1} \left\| \Delta \boldsymbol{\tau}(k+j|k) \right\|_R^2$$

$$\text{s.t.} \begin{bmatrix} \dot{\boldsymbol{\theta}}^{k_i} \\ \ddot{\boldsymbol{\theta}}^{k_i} \end{bmatrix} = \begin{bmatrix} \mathbf{0} & I_{2\times 2} \\ \mathbf{0} & -\mathbf{D}^{-1}(\boldsymbol{\theta}^{k_i})\mathbf{C}(\boldsymbol{\theta}^{k_i}, \dot{\boldsymbol{\theta}}^{k_i}) \end{bmatrix} \begin{bmatrix} \boldsymbol{\theta}^{k_i} \\ \dot{\boldsymbol{\theta}}^{k_i} \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \mathbf{D}^{-1}(\boldsymbol{\theta}^{k_i}) \end{bmatrix} \boldsymbol{\tau}^{k_j}$$

$$- \begin{bmatrix} \mathbf{0} \\ \mathbf{D}^{-1}(\boldsymbol{\theta}^{k_i})\mathbf{G}(\boldsymbol{\theta}^{k_i}) \end{bmatrix},$$

$$\boldsymbol{\theta}(k+i|k) \in [\boldsymbol{\theta}_{\min}, \boldsymbol{\theta}_{\max}], \dot{\boldsymbol{\theta}}(k+i|k) \in [\dot{\boldsymbol{\theta}}_{\min}, \dot{\boldsymbol{\theta}}_{\max}],$$

$$\boldsymbol{\tau}(k+j|k) \in [\boldsymbol{\tau}_{\min}, \boldsymbol{\tau}_{\max}],$$

$$i = 1, 2, \ldots, N, j = 1, 2, \ldots, N_\tau,$$

(37)

where $k_{i,j} \triangleq k + i, j|k$, and $\boldsymbol{\theta}^k$, $\dot{\boldsymbol{\theta}}^k$, and $\boldsymbol{\tau}^k$ represent the angle vector, angular velocity vector, and torque vector of two-link lower limb rehabilitation robot at prediction horizon and control horizon, respectively. $\boldsymbol{\theta}^{\mathrm{d}}(k+i|k)$ means the desired trajectory at $k+i$ time, and $\Delta \boldsymbol{\tau}(k+j|k)$ is the control input increment. $Q = 5I$ and $R = I$, where I is the identity matrix and the index $N = 3$ and $N_\tau = 3$.

The trained ESN model can effectively identify the joint angle and angular velocity of injured lower limb from the multichannel sEMG signals. The lower limb rehabilitation robot takes the results of recognition as the desired trajectories. Combining the ESN model and MPC technique, the human–machine interactive control method is developed, investigated, and analyzed for lower limb rehabilitation robot and injured lower limb in this paper. Besides, to design the human-machine interactive controller, the joint angle and angular velocity that can be regarded as desired trajectories are identified by ESN learning algorithm from 32 to 46 s. The numerical results are shown in **Figure 16**.

Figures 16A–C represent the numerical results of angles, angular velocities, and torques of lower-limb rehabilitation robot during rehabilitation training process. As shown in **Figure 16**, the blue solid lines mean the results of knee and the red solid lines represent the results of ankle, respectively. In light of **Figure 16**, it can be seen that the patients can be stably driven to do rehabilitation motion by lower limb rehabilitation robot, which be oriented by a human's active intention. It is also

inferred that the rehabilitation robot appropriately generates torques, which assist patients to do rehabilitation training and avoid the second injury. It is thus further demonstrated that it is very practical to train the injured lower limb through a human–machine interactive control method with multichannel sEMG signals. In other words, it is also verified that the ESN learning algorithm and Algorithm 1 (PASHS) are feasible and effective for the rehabilitation training of injured lower limb.

5. CONCLUSIONS

In this paper, to obtain an optimal controller of a non-linear system, an MPC problem firstly solved by a new PASHS algorithm has been proposed and analyzed by exploiting the three-order Taylor discretization formula to linearize and discretize the constraint conditions. Furthermore, the PASHS approach not only takes advantage of a projected operator, but it also integrates the active set into HS conjugate gradient methods; the optimal controller can thus be rapidly solved for a non-linear optimization problem. Moreover, the feasibility and global convergence have been rigorously proved in this paper. Some numerical results have been presented and analyzed to substantiate the feasibility, effectiveness, and superiority of the developed human–machine interactive control method for passive/active rehabilitation training. The ESN model with multichannel sEMG signals also has been proposed for intention recognition, which could identify the joint angles and angular velocities of the injured lower limb to realize active rehabilitation training. In other words, passive rehabilitation makes patients train through fixed-based trajectories of injured lower limb; however, the desired trajectories of active rehabilitation training are identified by ESN learning algorithm with multichannel signals. Besides, combining with MPC technology and Algorithm 1 (PASHS), human-machine interactive control has been developed, investigated, and analyzed for two-link lower limb rehabilitation robot. The numerical results have inferred that the proposed method could be effectively applied to passive/active rehabilitation training. The proposed method has also solved a

problem that creates uncertainty in the model. In future work, more effective and real-time methods will be developed and investigated in the solution of MPC problem and applied to the rehabilitation of patients, such as upper limb rehabilitation training, assisting patients to walk on the plane, or up and down stairs.

DATA AVAILABILITY STATEMENT

Publicly available datasets were analyzed in this study. This data can be found at: http://www.dqxy.ccut.edu.cn/2017/0906/c5519a78133/page.htm.

ETHICS STATEMENT

Ethical review and approval was not required for the study on human participants in accordance with the local legislation and institutional requirements. Written informed consent for participation was not required for this study in accordance with the national legislation and the institutional requirements.

AUTHOR CONTRIBUTIONS

TS: methodology, software, and review. YT: investigation and writing original draft. ZS: conceptualization and methodology. BZ: methodology. ZP: algorithm. JY: methodology and review. XZ: editing. All authors contributed to the article and approved the submitted version.

FUNDING

REFERENCES

Buchanan, T., Lloyd, D., Manal, K., and Besier, T. (2004). Neuromusculoskeletal modeling: estimation of muscle forces and joint moments and movements from measurements of neural command. *J. Appl. Biomech.* 20, 367–395. doi: 10.1123/jab.20.4.367

Cheng, W., Liu, Q., and Li, D. (2014). An accurate active set conjugate gradient algorithm with project search for bound constrained optimization. *Optim. Lett.* 8, 763–776. doi: 10.1007/s11590-013-0609-6

Dai, Y., and Yuan, Y. (2000a). A nonlinear conjugate gradient method with a strong global convergence property. *SIAM J. Optim.* 10, 177–182. doi: 10.1137/S1052623497318992

Dai, Y., and Yuan, Y. (2000b). *Nonlinear Conjugate Gradient Methods*. Shanghai: Shanghai Science and Technology Publisher.

Dai, Z. (2011). Two modified HS type conjugate gradient methods for unconstrained optimization problems. *Nonlin. Anal.* 74, 927–936. doi: 10.1016/j.na.2010.09.046

Dai, Z. (2014). Extension of modified Polak-Ribière-Polyak conjugate gradient method to linear equality constraints minimization problems. *Abstr. Appl. Anal.* 2014, 1–9. doi: 10.1155/2014/921364

Ding, Q., Xiong, A., Zhao, X., and Han, J. (2016). A review on researches and applications of semg-based motion intent recognition methods. *Acta Autom. Sin.* 42, 13–25. doi: 10.16383/j.aas.2016.c140563

Fleischer, C., and Hommel, G. (2008). A human-exoskeleton interface utilizing electromyography. *IEEE Trans. Robot.* 24, 872–882. doi: 10.1109/TRO.2008.926860

Fletcher, R., and Reeves, C. M. (1964). Function minimization by conjugate gradients. *Comput. J.* 7, 149–154. doi: 10.1093/comjnl/7.2.149

Guo, D., Nie, Z., and Yan, L. (2017). Novel discrete-time zhang neural network for time-varying matrix inversion. *IEEE Trans. Syst. Man Cybernet. Syst.* 47, 2301–2310. doi: 10.1109/TSMC.2017.2656941

Han, J., Ding, Q., Xiong, A., and Zhao, X. (2015). A state-space EMG model for the estimation of continuous joint movements. *IEEE Trans. Ind. Electron.* 62, 4267–4275. doi: 10.1109/TIE.2014.2387337

He, W., Ge, S. S., Li, Y., Chew, E., and Ng, Y. S. (2015). Neural network control of a rehabilitation robot by state and output feedback. *J. Intell. Robot. Syst.* 80, 15–31. doi: 10.1007/s10846-014-0150-6

Hestenes, M. R., and Stiefel, E. L. (1952). Methods of conjugate gradients for solving linear systems. *J. Res. Natl. Bureau Stand.* 49, 409–432. doi: 10.6028/jres.049.044

Hunt, K., Munih, M., Donaldson, N., and Barr, F. (1998). Investigation of the hammerstein hypothesis in the modeling of electrically stimulated muscle. *IEEE Trans. Biomed. Eng.* 45, 998–1009. doi: 10.1109/10.704868

Jin, D., and Zhang, J. (2011). *Bio-mechanology in Rehabilitation Engineering*. Beijing: Tsinghua University Press.

Jin, L., and Li, S. (2018). Distributed task allocation of multiple robots: a control perspective. *IEEE Trans. Syst. Man Cybernet. Syst.* 48, 693–701. doi: 10.1109/TSMC.2016.2627579

Jin, L., Li, S., Hu, B., Liu, M., and Yu, J. (2019). A noise-suppressing neural algorithm for solving the time-varying system of linear equations: a control-based approach. *IEEE Trans. Ind. Inform.* 15, 236–246. doi: 10.1109/TII.2018.2798642

Jin, L., Li, S., La, H. M., and Luo, X. (2017). Manipulability optimization of redundant manipulators using dynamic neural networks. *IEEE Trans. Ind. Electron.* 64, 4710–4720. doi: 10.1109/TIE.2017.2674624

Jin, L., and Zhang, Y. (2014). Discrete-time zhang neural network of $o\left(\tau^3\right)$ pattern for time-varying matrix pseudoinversion with application to manipulator motion generation. *Neurocomputing* 142, 165–173. doi: 10.1016/j.neucom.2014.04.051

Kanzow, C., and Klug, A. (2006). On affine-scaling interior-point newton methods for nonlinear minimization with bound constraints. *Comput. Optimiz. Appl.* 35, 177–197. doi: 10.1007/s10589-006-6514-5

Law, L. F., Krishnan, C., and Avin, K. (2011). Modeling nonlinear errors in surface electromyography due to baseline noise: a new methodology. *J. Biomech.* 44, 202–205. doi: 10.1016/j.jbiomech.2010.09.008

Li, C., and Li, D. (2013). An extension of the fletcher-reeves method to linear equality constrained optimization problem. *Appl. Math. Comput.* 219, 10909–10914. doi: 10.1016/j.amc.2013.04.055

Li, Q., Song, Y., and Hou, Z. (2015). Estimation of lower limb periodic motions from semg using least squares support vector regression. *Neural Process. Lett.* 41, 371–388. doi: 10.1007/s11063-014-9391-4

Liu, Y., and Storey, C. (1991). Efficient generalized conjugate gradient algorithms, part 1: theory. *J. Optimiz. Theory Appl.* 69, 129–137. doi: 10.1007/BF00940464

Mayne, D. Q. (2014). Model predictive control: recent developments and future promise. *Automatica* 50, 2967–2986. doi: 10.1016/j.automatica.2014.10.128

Pan, Y., and Wang, J. (2012). Model predictive control of unknown nonlinear dynamical systems based on recurrent neural networks. *IEEE Trans. Ind. Electron.* 59, 3089–3101. doi: 10.1109/TIE.2011.2169636

Pehlivan, A. U., Losey, D. P., and O'Malley, M. K. (2016). Minimal assist-as-needed controller for upper limb robotic rehabilitation.

IEEE Trans. Robot. 32, 113–124. doi: 10.1109/TRO.2015.25 03726

Peng, L., Hou, Z., Wang, C., Luo, L., and Wang, W. (2018). Physical interaction methods for rehabilitation and assistive robots. *Acta Autom. Sin.* 44, 2000–2010. doi: 10.16383/j.aas.2018.c180209

Polak, E., and Ribière, G. (1969). Note sur la convergence de méthodes de directions conjuguées. *Rev. Franç. Inform. Rec. Opér.* 16, 35–43. doi: 10.1051/m2an/196903R100351

Polyak, B. (1969). The conjugate gradient method in extreme problems. *USSR Comput. Math. Math. Phys.* 9, 94–112. doi: 10.1016/0041-5553(69)90035-4

Qi, Y., Jin, L., Li, H., Li, Y., and Liu, M. (2020). Discrete computational neural dynamics models for solving time-dependent sylvester equations with applications to robotics and mimo systems. *IEEE Trans. Ind. Inform.* 16, 6231–6241. doi: 10.1109/TII.2020.2966544

Qi, Y., Jin, L., Wang, Y., Xiao, L., and Zhang, J. (2019). Complex-valued discrete-time neural dynamics for perturbed time-dependent complex quadratic programming with applications. *IEEE Trans. Neural Netw. Learn. Syst.* 31, 3555–3569. doi: 10.1109/TNNLS.2019.2944992

Rosen, J. B. (1960). The gradient projection method for nonlinear programming. I. Linear constraints. *J. Soc. Ind. Appl. Math.* 8, 181–217. doi: 10.1137/0108011

Šantin, O., and Havlena, V. (2011). "Combined partial conjugate gradient and gradient projection solver for MPC," in *2011 IEEE International Conference on Control Applications (CCA)* (Denver, CO), 1270–1275. doi: 10.1109/CCA.2011.6044405

Sun, Z., Sun, Y., Li, Y., and Liu, K. (2019). A new trust region-sequential quadratic programming approach for nonlinear systems based on nonlinear model predictive control. *Eng. Optim.* 51, 1071–1096. doi: 10.1080/0305215X.2018.1509960

Sun, Z., Tian, Y., and Wang, J. (2018). A novel projected Fletcher-Reeves conjugate gradient approach for finite-time optimal robust controller of linear constraints optimization problem: application to bipedal walking robots. *Optim. Control Appl. Methods* 39, 130–159. doi: 10.1002/oca.2339

Sun, Z. B., Zhang, B. C., Sun, Y., Pang, Z. X., and Cheng, C. (2020). A novel superlinearly convergent trust region-sequential quadratic programming approach for optimal gait of bipedal robots via nonlinear model predictive control. *J. Intell. Robot. Syst.* 100, 401–416. doi: 10.1007/s10846-020-01174-4

Tong, L., Hou, Z., Peng, L., Chen, W. W. Y., and Tan, M. (2014). Multi-channel semg time series analysis based human motion recognition method. *Acta Autom. Sin.* 40, 810–821. doi: 10.3724/SP.J.1004.2014.00810

Wei, L., Jin, L., Yang, C., Chen, K., and Li, W. (2019). New noise-tolerant neural algorithms for future dynamic nonlinear optimization with estimation

on Hessian matrix inversion. *IEEE Trans. Syst. Man Cybernet. Syst.* 1–13. doi: 10.1109/TSMC.2019.2916892

Xie, Z., Jin, L., Du, X., Xiao, X., Li, H., and Li, S. (2019). On generalized rmp scheme for redundant robot manipulators aided with dynamic neural networks and nonconvex bound constraints. *IEEE Trans. Ind. Inform.* 15, 5172–5181. doi: 10.1109/TII.2019.2899909

Xie, Z., Jin, L., Luo, X., Li, S., and Xiao, X. (2020). A data-driven cyclic-motion generation scheme for kinematic control of redundant manipulators. *IEEE Trans. Control Syst. Technol.* 1–11. doi: 10.1109/TCST.2019.2963017

Yan, Z., and Wang, J. (2014). Robust model predictive control of nonlinear systems with unmodeled dynamics and bounded uncertainties based on neural networks. *IEEE Trans. Neural Netw. Learn. Syst.* 25, 457–469. doi: 10.1109/TNNLS.2013.2275948

Zhang, F., Li, P., Hou, Z., Lu, Z., Chen, Y., Li, Q., and Tan, M. (2012). sEMG-based continuous estimation of joint angles of human legs by using BP neural network. *Neurocomputing* 78, 139–148. doi: 10.1016/j.neucom.2011.05.033

Zhang, J., and Cheah, C. (2015). Passivity and stability of human-robot interaction control for upper-limb rehabilitation robots. *IEEE Trans. Robot.* 31, 233–245. doi: 10.1109/TRO.2015.2392451

Zhang, J., Jin, L., and Cheng, L. (2020). RNN for perturbed manipulability optimization of manipulators based on a distributed scheme: a game-theoretic perspective. *IEEE Trans. Neural Netw. Learn. Syst.* 1–11. doi: 10.1109/TNNLS.2020.2963998

Zhang, Y., Qi, Z., Li, J., Qiu, B., and Yang, M. (2019). Stepsize domain confirmation and optimum of ZeaD formula for future optimization. *Numer. Algorithms* 81, 561–574. doi: 10.1007/s11075-018-0561-8

Zorowitz, R. D., Gillard, P. J., and Brainin, M. (2013). Post stroke spasticity: sequelae and burden on stroke survivors and caregivers. *Neurology* 80, S45–S52. doi: 10.1212/WNL.0b013e3182764c86

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

APPENDIX

Symbols of the Model and Algorithm

$\mathbf{x}^k$	system state at kth sampling instant
$\mathbf{u}^k$	control input at kth sampling instant
$\mathbf{y}^k$	predicted output at kth sampling instant
$\mathbf{r}^k$	desired output at kth sampling instant
Γ	objective funtion
$\nabla\Gamma$	gradient of objective function
Λ, b	coefficient matrix and vector of equality constraint
Ω	bound constrained set
$H(x)$ or H^*	active set
$L(x)$ or L^*	free variables set
$P_\Omega(x)$	projection function
$\mathscr{P}$	projection matrix
d^k	search direction of algorithm
η^k	step size of Armijio-type line search rule
$\mathcal{X}$	echo state of ESN
$\mathcal{U}, \mathcal{Y}$	inputs and outputs of ESN
$\mathfrak{M}_\mathcal{X}, \mathfrak{M}_\mathcal{U}, \mathfrak{M}_\mathcal{F}, \mathfrak{M}_\mathcal{Y},$	matrixes of ESN

Sensor-Based Control for Collaborative Robots: Fundamentals, Challenges, and Opportunities

Andrea Cherubini[1]* and David Navarro-Alarcon[2]

[1] LIRMM, Univ Montpellier, CNRS, Montpellier, France, [2] Department of Mechanical Engineering, The Hong Kong Polytechnic University, Hong Kong, Hong Kong

The objective of this paper is to present a systematic review of existing sensor-based control methodologies for applications that involve direct interaction between humans and robots, in the form of either physical collaboration or safe coexistence. To this end, we first introduce the basic formulation of the sensor-servo problem, and then, present its most common approaches: vision-based, touch-based, audio-based, and distance-based control. Afterwards, we discuss and formalize the methods that integrate heterogeneous sensors at the control level. The surveyed body of literature is classified according to various factors such as: sensor type, sensor integration method, and application domain. Finally, we discuss open problems, potential applications, and future research directions.

Keywords: robotics, human-robot collaboration (HRC), human-robot interaction (HRI), control systems (CS), visual servoing (VS)

1. INTRODUCTION

Robot control is a mature field: one that is already being heavily commercialized in industry. However, the methods required to regulate interaction and collaboration between humans and robots have not been fully established yet. These issues are the subject of research in the fields of physical human-robot interaction (pHRI) (Bicchi et al., 2008) and collaborative robotics (CoBots) (Colgate et al., 1996). The authors of De Luca and Flacco (2012) presented a paradigm that specifies three nested layers of consistent behaviors that the robot must follow to achieve safe pHRI:

- *Safety* is the first and most important feature in collaborative robots. Although there has been a recent push toward standardization of robot safety (e.g., the ISO 13482:2014 for robots and robotic devices; ISO 13482:2014, 2014), we are still in the initial stages. Safety is generally addressed through *collision avoidance* (with both humans or obstacles; Khatib, 1985), a feature that requires high reactivity (high bandwidth) and robustness at both the perception and control layers.
- *Coexistence* is the robot capability of sharing the workspace with humans. This includes applications involving a passive human (e.g., medical operations where the robot is intervening on the patients' body; Azizian et al., 2014), as well as scenarios where robot and human work together on the same task, without contact or coordination.
- *Collaboration* is the capability of performing robot tasks with direct human interaction and coordination. There are two modes for this: *physical* collaboration (with explicit and intentional contact between human and robot), and *contactless* collaboration (where the actions are guided by an exchange of information, e.g., in the form of body gestures, voice commands, or other modalities). Especially for the second mode, it is crucial to establish means for intuitive control by the human operators, which may be non-expert users. The robot should be proactive in

realizing the requested tasks, and it should be capable of inferring the user's intentions, to interact more naturally from the human viewpoint.

All three layers are hampered by the unpredictability of human actions, which vary according to situations and individuals, complicating modeling (Phoha, 2014), and use of classic control.

In the robotics literature, two major approaches for task execution have emerged: *path/motion planning* (La Valle, 2006) and *sensor-based control* (Chaumette and Hutchinson, 2006). The *planning* methods rely on a priori knowledge of the future robot and environment states over a time window. Although they have proved their efficiency in well-structured applications, these methods are hardly applicable to human-robot collaboration, because of the unpredictable and dynamic nature of humans. It is in the authors' view that *sensor-based control* is more efficient and flexible for pHRI, since it closes the perception-to-action loop at a lower level than path/motion planning. Note also that sensor-based control strategies strongly resemble the processes of our central nervous system (Berthoz, 2002), and can trace their origins back to the servomechanism problem (Davison and Goldenberg, 1975). The most known example is image-based visual servoing (Chaumette and Hutchinson, 2006) which relies directly on visual feedback to control robot motion, without requiring a cognitive layer nor a precise model of the environment.

The aim of this article is to survey the current state of art in *sensor-based control, as a means to facilitate the interaction between robots, humans, and surrounding environments.* Although we acknowledge the need for other techniques within a complete human-robot collaboration framework (e.g., path planning as mentioned, machine learning, etc.), here we review and classify the works which exploit sensory feedback to directly command the robot motion.

The timing and relevance of this survey is twofold. On one hand, while there have been previous reviews on topics such as (general) human-robot collaboration (Ajoudani et al., 2017; Villani et al., 2018) and human-robot safety (Haddadin et al., 2017), there is no specific review on the use of *sensor-based control* for human-robot collaborative tasks. On the other hand, we introduce a *unifying* paradigm for designing controllers with four sensing modalities. This feature gives our survey a valuable tutorial-like nature.

The rest of this manuscript is organized as follows: Section 2 presents the basic formulation of the sensor-based control problem; Section 3 describes the common approaches that integrate multiple sensors at the control level. Section 4 provides several classifications of the reviewed works. Section 5 presents insights and discusses open problems and areas of opportunity. Conclusions are given in section 6.

2. SENSING MODALITIES FOR CONTROL

Recent developments on bio-inspired measurement technologies have made sensors affordable and lightweight, easing their use on robots. These sensors include RGB-D cameras, tactile skins, force/moment transducers, etcetera (see **Figure 1**). The works reviewed here rely on different combinations of sensing modalities, depending on the task at stake. We consider the following four robot senses:

- *Vision.* This includes methods for processing and understanding images, to produce numeric or symbolic information reproducing human sight. Although image processing is complex and computationally expensive, the richness of this sense is unique. Robotic vision is fundamental for understanding the environment and human intention, so as to react accordingly.
- *Touch.* In this review, touch includes both *proprioceptive force* and *tact*, with the latter involving *direct* physical contact with an external object. *Proprioceptive force* is analogous to the sense of muscle force (Proske and Gandevia, 2012). The robot can measure it either from the joint position errors or via torque sensors embedded in the joints; it can then use both methods to infer and adapt to human intentions, by relying on force control (Raibert and Craig, 1981; Hogan, 1985; Morel et al., 1998; Villani and De Schutter, 2008). Human *tact* (somatosensation), on the other hand, results from activation of neural receptors, mostly in the skin. These have inspired the design of artificial tactile skins (Wettels et al., 2008; Schmitz et al., 2011), thoroughly used for human-robot collaboration.
- *Audition.* In humans, localization of sound is performed by using binaural audition (i.e., two ears). By exploiting auditory cues in the form of level/time/phase differences between left and right ears we can determine the source's horizontal position and its elevation (Rayleigh, 1907). Microphones artificially emulate this sense, and allow robots to "blindly" locate sound sources. Although robotic hearing typically uses two microphones mounted on a motorized head, other non-biological configurations exist, e.g., a head instrumented with a single microphone or an array of several omni-directional microphones (Nakadai et al., 2006).
- *Distance.* This is the only sense among the four that humans cannot directly measure. Yet, numerous examples exist in the mammal kingdom (e.g., bats and whales), in the form of echolocation. Robots measure distance with *optical* (e.g., infrared or lidar), *ultrasonic*, or *capacitive* (Göger et al., 2010) sensors. The relevance of this particular "sense" in human-robot collaboration is motivated by the direct relationship existing between the distance from obstacles (here, the human) and safety.

Roboticists have designed other bio-inspired sensors, to *smell* (see Kowadlo and Russell, 2008 for a comprehensive survey and Russell, 2006; Gao et al., 2016; Rahbar et al., 2017 for 3D tracking applications) and *taste* (Shimazu et al., 2007; Kobayashi et al., 2010; Ha et al., 2015). However, in our opinion, artificial *smell* and *taste* are not yet mature enough for human-robot collaboration. Most of the current work on these senses is for localization/identification of hazardous gases/substances. It is also worth mentioning the increasing popularity of physiological signals for controlling robots. These include, for example, Electromyography and Brain-Computer Interfaces (Ajoudani et al., 2017). Albeit promising, these technologies generally provide unilateral (from human to robot) control, without

FIGURE 1 | Examples of artificial sensors. Clockwise from the top left: Microsoft Kinect® and Intel Realsense® (vision and distance), Sony D-Link DCS-5222L® and AVT GT® (vision), Syntouch BioTac® and ATI Nano 43® (touch), sound sensor LM393® and 3Dio Free Space Pro II® Binaural Microphone (audition), proximity sensor Sharp GP2Y0A02YK0F®, Laser SICK®, Hokuyo URG®, and proximity sensor SICK CM18-08BPP-KC1® (distance). Note that Intel Realsense® and Microsoft Kinect® provide both the senses of vision and of distance.

feedback loop closure. For these reasons, this review will focus only on the four senses mentioned above, namely vision, touch, audition, and distance.

3. SENSOR-BASED CONTROL

3.1. Formulation of Sensor-Based Control

Sensor-based control aims at deriving the robot control input $\mathbf{u}$ (operational space velocity, joint velocity, displacement, etc.) that minimizes a *trajectory* error $\mathbf{e} = \mathbf{e}(\mathbf{u})$, which can be estimated by sensors and depends on $\mathbf{u}$. A general way of formulating this controller [accounting for actuation redundancy $\dim(\mathbf{u}) > \dim(\mathbf{e})$, sensing redundancy $\dim(\mathbf{u}) < \dim(\mathbf{e})$, and task constraints] is as the quadratic minimization problem:

$$\mathbf{u} = \arg\min_{\mathbf{u}} \frac{1}{2} \|\mathbf{e}(\mathbf{u})\|^2 \tag{1}$$

subject to task constraints.

This formulation encompasses the classic *inverse kinematics* problem (Whitney, 1969) of controlling the robot joint velocities ($\mathbf{u} = \dot{\mathbf{q}}$), so that the end-effector operational space position $\mathbf{x}$ converges to a desired value $\mathbf{x}^*$. By defining the desired end-effector rate as $\dot{\mathbf{x}}^* = -\lambda(\mathbf{x} - \mathbf{x}^*)$, for $\lambda > 0$, and setting $\mathbf{e} = \mathbf{J}\dot{\mathbf{q}} - \dot{\mathbf{x}}^*$ for $\mathbf{J} = \partial\mathbf{x}/\partial\mathbf{q}$ as the Jacobian matrix, it is easy to show that the solution to (1) (in the absence of constraints) is $\dot{\mathbf{q}} = \mathbf{J}^+\dot{\mathbf{x}}^*$, with $\mathbf{J}^+$ the generalized inverse of $\mathbf{J}$. This leads to the set-point controller[1]:

---

[1]Throughout the paper, λ is a positive tuning scalar that determines the convergence rate of task error $\mathbf{e}$ to 0.

$$\dot{\mathbf{q}} = -\mathbf{J}^+\lambda(\mathbf{x} - \mathbf{x}^*). \tag{2}$$

In the following sections, we show how each of the four senses (*vision*, *touch*, *audition* and *distance*) has been used for robot control, either with (1), or with similar techniques. **Figure 2** shows relevant variables for the four cases. For simplicity, we assume there are no constraints in (1), although off-the-shelf quadratic programming solvers (Nocedal and Wright, 2000) could account for them.

3.2. Visual Servoing
3.2.1. Formulation
Visual servoing refers to the use of vision to control the robot motion (Chaumette and Hutchinson, 2006). The camera may be mounted on a moving part of the robot, or fixed in the workspace. These two configurations are referred to as "eye-in-hand" and "eye-to-hand" visual servoing, respectively. The error $\mathbf{e}$ is defined with regards to some image features, here denoted by $\mathbf{s}$, to be regulated to a desired configuration $\mathbf{s}^*$ ($\mathbf{s}$ is analogous to $\mathbf{x}$ in the inverse kinematic formulation above). The visual error is:

$$\mathbf{e} = \dot{\mathbf{s}} - \dot{\mathbf{s}}^*. \tag{3}$$

Visual servoing schemes are called *image-based* if $\mathbf{s}$ is defined in image space, and *position-based* if $\mathbf{s}$ is defined in the 3D operational space. Here we only briefly recall the image-based approach (on its eye-in-hand modality), since the position-based one consists in projecting the task from the image to the operational space to obtain $\mathbf{x}$ and then apply (2).

The simplest image-based controller uses $\mathbf{s} = [X, Y]^\top$, with X and Y as the coordinates of an image pixel, to generate $\mathbf{u}$ that

FIGURE 2 | Examples of four sensor-based servo controllers. **(A)** Visual servoing: the user hand is centered in the camera image. **(B)** Indirect force control: by applying a wrench, the user deviates the contact point away from a reference trajectory. **(C)** Audio-based control: a microphone rig is automatically oriented toward the sound source (the user's mouth). **(D)** Distance-based control: the user acts as a repulsive force, related to his/her distance from the robot.

drives **s** to a reference $\mathbf{s}^* = [X^*, Y^*]^\top$ (in **Figure 2A** the centroid of the human hand). This is done by defining **e** as:

$$\dot{\mathbf{s}} - \dot{\mathbf{s}}^* = \begin{bmatrix} \dot{X} - \dot{X}^* \\ \dot{Y} - \dot{Y}^* \end{bmatrix}, \text{with } \dot{\mathbf{s}}^* = -\lambda \begin{bmatrix} X - X^* \\ Y - Y^* \end{bmatrix} \quad (4)$$

If we use the camera's 6D velocity as the control input $\mathbf{u} = \mathbf{v}_c$, the image Jacobian matrix[2] relating $[\dot{X}, \dot{Y}]^\top$ and **u** is:

$$\mathbf{J}_v = \begin{bmatrix} -\frac{1}{\zeta} & 0 & \frac{X}{\zeta} & XY & -1-X^2 & Y \\ 0 & -\frac{1}{\zeta} & \frac{Y}{\zeta} & 1+Y^2 & -XY & -X \end{bmatrix}, \quad (5)$$

where ζ denotes the depth of the point with respect to the camera. In the absence of constraints, the solution of (1) is:

$$\mathbf{v}_c = -\mathbf{J}_v^+ \lambda \begin{bmatrix} X - X^* \\ Y - Y^* \end{bmatrix}. \quad (6)$$

3.2.2. Application to Human-Robot Collaboration

Humans generally use vision to teach the robot relevant configurations for collaborative tasks. For example, Cai et al. (2016) demonstrate an application where a human operator used a QR code to specify the target poses for a 6 degrees-of-freedom (dof) robot arm. In Gridseth et al. (2016), the user provided target tasks via a tablet-like interface that sent the robot the desired reference view; here, the human specified

---

[2] Also known as *interaction matrix* in the visual servoing literature.

various motions such as point-to-point, line-to-line, etc., that were automatically performed via visual feedback. The authors of Gridseth et al. (2015) presented a grasping system for a tele-operated dual arm robot, where the user specified the object to be manipulated, and the robot completed the task using visual servoing.

Assistive robotics represents another very common application domain for visual servoing. The motion of robotic wheelchairs has been semi-automated at various degrees. For instance, Narayanan et al. (2016) presented a corridor following method that exploited the projection of parallel lines. In this work, the user provided target directions with a haptic interface, and the robot corrected the trajectories with visual feedback. Other works have focused on mobile manipulation. The authors of Tsui et al. (2011) developed a vision-based controller for a robotic arm mounted on a wheelchair; in this work, the user manually specified the object to be grasped and retrieved by the robot. A similar approach was reported in Dune et al. (2008), where the desired poses were provided with "clicks" on an screen interface.

Medical robotics is another area that involves sensor-based interactions between humans and robots, and where vision has huge potential (see Azizian et al., 2014 for a comprehensive review). For example, the authors of Agustinos et al. (2014) developed a laparoscopic camera, which regulated its pan/tilt motions to track human-held instruments.

3.3. Touch (or Force) Control
3.3.1. Formulation

Touch (or force) control requires the measurement of one or multiple (in the case of tactile skins) *wrenches* **h**, which are (at most) composed of three translational forces, and three torques; **h** is fed to the controller that moves the robot so that it exerts a desired interaction force with the human or environment. Force control strategies can be grouped into the following two classes (Villani and De Schutter, 2008):

- *Direct* control regulates the contact wrench to obtain a desired wrench $\mathbf{h}^*$. Specifying $\mathbf{h}^*$ requires an explicit model of the task and environment. A widely adopted strategy is hybrid position/force control (Raibert and Craig, 1981), which regulates the velocity and wrench along unconstrained and constrained task directions, respectively. Referring to (1), this is equivalent to setting

$$\mathbf{e} = \mathbf{S}\left(\dot{\mathbf{x}} - \dot{\mathbf{x}}^*\right) + \left(\mathbf{I} - \mathbf{S}\right)\left(\mathbf{h} - \mathbf{h}^*\right), \quad (7)$$

with $\mathbf{S} = \mathbf{S}^\top \geq 0$ a binary diagonal selection matrix, and **I** as the identity matrix. Applying a motion **u** that nullifies **e** in (7) guarantees that the components of $\dot{\mathbf{x}}$ (respectively **h**) specified via **S** (respectively $\mathbf{I} - \mathbf{S}$) converge to $\dot{\mathbf{x}}^*$ (respectively $\mathbf{h}^*$).

- *Indirect* control (illustrated in **Figure 2B**) does not require an explicit force feedback loop. To this category belong *impedance control* and its dual *admittance control* (Hogan, 1985). It consists in modeling the deviation of the contact point from a reference trajectory $\mathbf{x}^r(t)$ associated to the desired $\mathbf{h}^*$, via a virtual mechanical impedance with adjustable

parameters (inertia $\mathbf{M}$, damping $\mathbf{B}$, and stiffness $\mathbf{K}$). Referring to (1), this is equivalent to setting:

$$\mathbf{e} = \mathbf{M}(\ddot{\mathbf{x}} - \ddot{\mathbf{x}}^r) + \mathbf{B}(\dot{\mathbf{x}} - \dot{\mathbf{x}}^r) + \mathbf{K}(\mathbf{x} - \mathbf{x}^r) - (\mathbf{h} - \mathbf{h}^*). \quad (8)$$

Here, $\mathbf{x}$ represents the "deviated" contact point pose, with $\dot{\mathbf{x}}$ and $\ddot{\mathbf{x}}$ as time derivatives. When $\mathbf{e} = \mathbf{0}$, the displacement $\mathbf{x} - \mathbf{x}^r$ responds as a mass-spring-damping system under the action of an external force $\mathbf{h} - \mathbf{h}^*$. In most cases, $\mathbf{x}^r(t)$ is defined for motion in free space ($\mathbf{h}^* = \mathbf{0}$). The general formulation in (1) and (8) can account for both impedance control ($\mathbf{x}$ is measured and $\mathbf{u} = \mathbf{h}$) and admittance control ($\mathbf{h}$ measured and $\mathbf{u} = \mathbf{x}$).

3.3.2. Application to Human-Robot Collaboration

The authors of Bauzano et al. (2016) used direct force control for collaborative human-robot laparoscopic surgery. In their method, the instruments are controlled with a hybrid position/force approach. In Cortesao and Dominici (2017), a robot regulated the applied forces onto a beating human heart. Since the end-effector's 3 linear dof were fully-constrained, position control could not be performed, i.e., $\mathbf{S} = \mathbf{0}$ in (7).

A drawback of direct control is that it can realize only the tasks which can be described via constraint surfaces. If their location is unknown and/or the contact geometry is complex—as often in human-robot collaboration—indirect control is more suited since: (i) it allows to define a priori how the robot should react to unknown external force disturbances, (ii) it can use a reference trajectory $\mathbf{x}^r(t)$ output by another sensor (e.g., vision). In the next paragraph, we review indirect force control methods.

By sensing force, the robot can infer the motion commands (e.g., pushing, pulling) from the human user. For example, Maeda et al. (2001) used force sensing and human motion estimation (based on minimum jerk) within an indirect (admittance) control framework for cooperative manipulation. In Suphi Erden and Tomiyama (2010) and Suphi Erden and Maric (2011), an assistant robot suppressed involuntary vibrations of a human, who controlled direction and speed of a welding operation. By exploiting kinematic redundancy, Ficuciello et al. (2013) also addressed a manually guided robot operation. The papers (Bussy et al., 2012; Wang et al., 2015) presented admittance controllers for two-arm robots moving a table in collaboration with a human. In Baumeyer et al. (2015), a human controlled a medical robot arm with an admittance controller. Robot *tele-operation* is another common human-robot collaboration application where force feedback plays a crucial role; see Passenberg et al. (2010) for a comprehensive review on the topic.

All these works relied on local force/moment measurements. Up to this date, tactile sensors and skins (measuring the wrench along the robot body, see Argall and Billard, 2010 for a review) have been used for object exploration (Natale and Torres-Jara, 2006) or recognition (Abderrahmane et al., 2018), but not for control as expressed in (1). One reason is that they are at a preliminary design stage, which still requires complex calibration (Del Prete et al., 2011; Lin et al., 2013) that constitutes a research topic per se. An exception is Li et al. (2013), which presented a method that used tactile measures within (1). Similarly, in Zhang and Chen (2000), tactile sensing was used

to regulate interaction with the environment. Yet, neither of these works considered pHRI. In our opinion, there is huge potential in the use of skins and tactile displays for human-robot collaboration.

3.4. Audio-Based Control

3.4.1. Formulation

The purpose of audio-based control is to locate the sound source, and move the robot toward it. For simplicity, we present the two-dimensional binaural (i.e., with two microphones) configuration in **Figure 2C**, with the angular velocity of the microphone rig as control input: $\mathbf{u} = \dot{\alpha}$. We hereby review the two most popular methods for defining error $\mathbf{e}$ in (1): Interaural Time Difference (ITD) and Interaural Level Difference (ILD)[3]. The following is based on Magassouba et al. (2016b):

- *ITD-based aural servoing* uses the difference τ between the arrival times of the sound on each microphone; τ must be regulated to a desired τ^*. The controller can be represented with (1), by setting $\mathbf{e} = \dot{\tau} - \dot{\tau}^*$, with the desired rate $\dot{\tau}^* = -\lambda(\tau - \tau^*)$ (to obtain set-point regulation to τ^*). Feature τ can be derived in real-time by using standard cross-correlation of the signals (Youssef et al., 2012). Under a far field assumption:

$$\mathbf{e} = \dot{\tau} - \dot{\tau}^* = -\left(\sqrt{(b/c)^2 - \tau^2}\right)\mathbf{u} - \dot{\tau}^* \quad (9)$$

with c the sound celerity and b the microphones baseline. From (9), the *scalar* ITD Jacobian is: $\mathbf{J}_\tau = -\sqrt{(b/c)^2 - \tau^2}$. The motion that minimizes $\mathbf{e}$ is:

$$\mathbf{u} = -\lambda \mathbf{J}_\tau^{-1}(\tau - \tau^*), \quad (10)$$

which is locally defined for $\alpha \in (0, \pi)$, to ensure that $|\mathbf{J}_\tau| \neq 0$.

- *ILD-based aural servoing* uses ρ, the difference in intensity between the left and right signals. This can be obtained in a time window of size N as $\rho = E_l/E_r$, where the $E_{l,r} = \sum_{n=0}^{N} \gamma_{l,r}[n]^2$ denote the signals' sound energies and the $\gamma_{l,r}[n]$ are the intensities at iteration n. To regulate ρ to a desired ρ^*, one can set $\mathbf{e} = \dot{\rho} - \dot{\rho}^*$ with $\dot{\rho}^* = -\lambda(\rho - \rho^*)$. Assuming spherical propagation and slowly varying signal:

$$\mathbf{e} = \dot{\rho} - \dot{\rho}^* = \frac{y_s(\rho + 1)b}{L_r^2}\mathbf{u} - \dot{\rho}^* \quad (11)$$

where y_s is the sound source frontal coordinate in the moving auditory frame, and L_r the distance between the right microphone and the source. From (11), the *scalar* ILD Jacobian is $\mathbf{J}_\rho = y_s(\rho + 1)b/L_r^2$. The motion that minimizes e is:

$$\mathbf{u} = -\lambda \mathbf{J}_\rho^{-1}(\rho - \rho^*) \quad (12)$$

where $\mathbf{J}_\rho^{-1}$ is defined for sources located in front of the rig. In contrast with ITD-servoing, here the source location (i.e., y_s and L_r) must be known or estimated.

---

[3]Or its frequency counterpart: Interaural Phase Difference (IPD).

While the methods above only control the angular velocity of the rig ($\mathbf{u} = \dot{\alpha}$), Magassouba extended both to also regulate 2D translations of a mobile platform (ITD in Magassouba et al., 2015, 2016c and ILD in Magassouba et al., 2016a).

3.4.2. Application to Human-Robot Collaboration

Due to the nature of this sense, audio-based controllers are mostly used in contact-less applications, to enrich other senses (e.g., force, distance) with sound, or to design intuitive human-robot interfaces.

Audio-based control is currently (in our opinion) an underdeveloped research area with great potential for human-robot collaboration, e.g., for tracking a speaker. Besides the cited works (Magassouba et al., 2015, 2016a,b,c), that closely followed the framework of section 3, others have formulated the problem differently. For example, the authors of Kumon et al. (2003, 2005) proposed a linear model to describe the relation between the pan motion of a robot head and the difference of intensity between its two microphones. The resulting controllers were much simpler than (10) and (12). Yet, their operating range was smaller, making them less robust than their—more analytical—counterparts.

3.5. Distance-Based Control

3.5.1. Formulation

The simplest (and most popular) distance-based controller is the artificial potential fields method (Khatib, 1985). Despite being prone to local minima, it has been thoroughly deployed both on manipulators and on autonomous vehicles for obstacle avoidance. Besides, it is acceptable that a collaborative robot stops (e.g., because of local minima) as long as it avoids the human user. The potential fields method consists in modeling each obstacle as a source of repulsive forces, related to the robot distance from the obstacle (see **Figure 2D**). All the forces are summed up resulting in a velocity in the most promising direction. Given $\mathbf{d}$, the position of the nearest obstacle in the robot frame, the original version (Khatib, 1985) consists in applying operational space velocity

$$\mathbf{u} = \begin{cases} \lambda \left(\frac{1}{\|\mathbf{d}\|} - \frac{1}{d_o} \right) \frac{\mathbf{d}}{\|\mathbf{d}\|^2} & \text{if } \|\mathbf{d}\| < d_o, \\ 0 & \text{otherwise.} \end{cases} \quad (13)$$

Here $d_o > 0$ is the (arbitrarily tuned) minimal distance required for activating the controller. Since the quadratic denominator in (13) yields abrupt accelerations, more recent versions adopt a linear behavior. Referring to (1), this can be obtained by setting $\mathbf{e} = \dot{\mathbf{x}} - \dot{\mathbf{x}}^*$ with $\dot{\mathbf{x}}^* = \lambda \left(1 - d_0/\|\mathbf{d}\| \right) \mathbf{d}$ as reference velocity:

$$\mathbf{e} = \dot{\mathbf{x}} - \lambda \left(1 - \frac{d_0}{\|\mathbf{d}\|} \right) \mathbf{d}. \quad (14)$$

By defining as control input $\mathbf{u} = \dot{\mathbf{x}}$, the solution to (1) is:

$$\mathbf{u} = \lambda \left(1 - \frac{d_0}{\|\mathbf{d}\|} \right) \mathbf{d}. \quad (15)$$

3.5.2. Application to Human-Robot Collaboration

Many works have used this (or similar) distance-based methods for pHRI. To avoid human-robot collisions, the authors of De

Santis et al. (2007) applied the controller (15) by estimating the distance $\mathbf{d}$ between a human head and a robot with vision. Recently, these approaches have been boosted by the advent of 3D vision sensors (e.g., the Microsoft Kinect and Intel RealSense), which enable both vision and distance control. The authors of Flacco et al. (2012) designed a Kinect-based distance controller (again, for human collision avoidance) with an expression similar to (15), but smoothed by a sigmoid.

Proximity servoing is a similar technique, which regulates—via capacitive sensors—the distance between the robot surface and the human. In Schlegl et al. (2013), these sensors modified the position and velocity of a robot arm when a human approached it, to avoid collisions. The authors of Bergner et al. (2017), Leboutet et al. (2016), and Dean-Leon et al. (2017) developed a new capacitive skin for a dual-arm robot. They designed a collision avoidance method based on an admittance model similar to (8), which relied on the joint torques (measured by the skin) to control the robot motion.

4. INTEGRATION OF MULTIPLE SENSORS

In section 3, we presented the most common sensor-based methods used for collaborative robots. Just like natural senses, artificial senses provide complementary information about the environment. Hence, to effectively perform a task, the robot should measure (and use for control) multiple feedback modalities. In this section, we review various methods for integrating multiple sensors in a unique controller.

Inspired by how humans merge their percepts (Ernst and Banks, 2002), researchers have traditionally fused heterogeneous sensors to estimate the state of the environment. This can be done in the sensors' Cartesian frames (Smits et al., 2008) by relying on an Extended Kalman Filter (EKF) (Taylor and Kleeman, 2006). Yet the sensors must be related to a single quantity, which is seldom the case when measuring different physical phenomena (Nelson and Khosla, 1996). An alternative is to use the sensed feedback directly in (1). This idea, proposed for position-force control in Raibert and Craig (1981) and extended to vision in Nelson et al. (1995), brings new challenges to the control design, e.g., sensor synchronization, task compatibility, and task representation. For instance, the designer should take care when transforming 6 D velocities or wrenches to a unique frame. This requires (when mapping from frame A to frame B) multiplication by

$$^B\mathbf{V}_A = \begin{bmatrix} ^B\mathbf{R}_A & \left[^B\mathbf{t}_A \right]_\times ^B\mathbf{R}_A \\ \mathbf{0}_3 & ^B\mathbf{R}_A \end{bmatrix} \quad (16)$$

for a velocity, and by $^B\mathbf{V}_A^\top$ for a wrench. In (16), $^B\mathbf{R}_A$ is the rotation matrix from A to B and $\left[^B\mathbf{t}_A \right]_\times$ the skew-symmetric matrix associated to translation $^B\mathbf{t}_A$.

According to Nelson et al. (1995), the three methods for combining N sensors within a controller are:

FIGURE 3 | The most common scheme for *shared* vision/touch (admittance) control, used in Morel et al. (1998), Agravante et al. (2013, 2014). The goal is to obtain desired visual features **s*** and wrench **h***, based on current image **I** and wrench **h**. The *outer* visual servoing loop based on error (3) outputs a reference velocity $\dot{\mathbf{x}}^r$ that is then deformed by the *inner* admittance control loop based on error (8), to obtain the desired robot position **x**.

- *Traded*: the sensors control the robot one at a time. Predefined conditions on the task trigger the switches:

$$
\mathbf{u} = \begin{cases} \arg\min_{\mathbf{u}} \|\mathbf{e}_1(\mathbf{u})\|^2 & \text{if } (condition\ 1) = \text{true,} \\ \vdots \\ \arg\min_{\mathbf{u}} \|\mathbf{e}_N(\mathbf{u})\|^2 & \text{if } (condition\ N) = \text{true.} \end{cases} \tag{17}
$$

- *Shared*: All sensors control the robot throughout operation. A common way is via nested control loops, as shown—for shared vision/touch control—in **Figure 3**. Researchers have used at most two loops, denoted o for outer and i for inner loop:

$$
\mathbf{u} = \arg\min_{\mathbf{u}} \|\mathbf{e}_i(\mathbf{u}, \mathbf{u}_o)\|^2 \tag{18}
$$

$$
\text{such that } \mathbf{u}_o = \arg\min_{\mathbf{u}_o} \|\mathbf{e}_o(\mathbf{u}_o)\|^2.
$$

In the example of **Figure 3**: $\mathbf{u} = \mathbf{x}$, $\mathbf{u}_o = \dot{\mathbf{x}}^r$, $\mathbf{e}_o = \mathbf{e_v}$ applying (3) and $\mathbf{e}_i = \mathbf{e_t}$ applying (8).

- *Hybrid*: the sensors act simultaneously, but on different axes of a predefined Cartesian *task-frame* (Baeten et al., 2003). The directions are selected by binary diagonal matrices $\mathbf{S}_j$, $j = 1, \ldots, N$ with the dimension of the task space, and such that $\sum_{j=1}^{N} \mathbf{S} = \mathbf{I}$:

$$
\mathbf{u} = \arg\min_{\mathbf{u}} \|\sum_{j=1}^{N} \mathbf{S}_j \mathbf{e}_j(\mathbf{u})\|^2. \tag{19}
$$

To express all $\mathbf{e}_j$ in the same task frame, one should apply $^B\mathbf{V}_A$ and/or $^B\mathbf{V}_A^\top$. Note the analogy between (19) and the hybrid position/force control framework (7).

We will use this classification to characterize the works reviewed in the rest of this Section.

4.1. Traded Control

The paper (Cherubini et al., 2016) presented a human-robot manufacturing cell for collaborative assembly of car joints. The approach (traded *vision/touch*) could manage physical contact between robot and human, and between robot and environment, via admittance control (8). Vision would take over

in dangerous situations to trigger emergency stops. The switching *condition* was determined by the position of the human wrt the robot.

In Okuno et al. (2001, 2004), a traded *vision/audio* controller enabled a mobile robot to exploit sound source localization for visual control. The robot head would automatically rotate toward the estimated direction of the human speaker, and then visually track him/her. The switching *condition* is that the sound source is visible. The audio-based task is equivalent to regulating τ to 0 or ρ to 1, as discussed in section 3.4. Paper (Hornstein et al., 2006) presented another traded *vision/audio* controller for the iCub robot head to localize a human speaker. This method constructed audio-motor maps and integrated visual feedback to update the map. Again, the switching *condition* is that the speaker's face is visible. In Chan et al. (2012), another traded *vision/audio* controller was deployed on a mobile robot, to drive it toward an unknown sound source; the switching *condition* is defined by a threshold on the frontal localization error.

The authors of Papageorgiou et al. (2014) presented a mobile assistant for people with walking impairments. The robot was equipped with: two wrench sensors to measure physical interaction with the human, an array of microphones for audio commands, laser sensors for detecting obstacles, and an RGB-D camera for estimating the users' state. Its controller integrated *audio*, *touch*, *vision*, and *distance* in a traded manner, with switching *conditions* determined by a knowledge-based layer.

The work (Navarro et al., 2014) presented an object manipulation strategy, integrating *distance* (capacitive proximity sensors) and *touch* (tactile sensors). While the method did not explicitly consider humans, it may be applied for human-robot collaboration, since proximity sensors can detect humans if vision is occluded. The switching *condition* between the two modes is the contact with the object.

Another example of traded control—here, *audio/distance*—is Huang et al. (1999), which presented a method for driving a mobile robot toward hidden sound sources, via an omnidirectional array of microphones. The controller switched to ultrasound-based obstacle avoidance in the presence of humans/objects. The detection of a nearby obstacle is the switching *condition*.

4.2. Shared Control

In applications where the robot and environment/human are in permanent contact (e.g., collaborative object transportation), shared control is preferable. Let us first review a pioneer controller (Morel et al., 1998) that relied on shared *vision/touch*, as outlined in **Figure 3**; Morel et al. (1998) addressed tele-operated peg-in-hole assembly, by placing the visual loop outside the force loop. The reference trajectory $\dot{\mathbf{x}}^r$ output by visual servoing was deformed in the presence of contact by the admittance controller, to obtain the robot position command $\mathbf{x}$. Human interaction was not considered in this work.

The authors of Natale et al. (2002) estimated sensory-motor responses to control a pan-tilt robot head with shared *visual/audio* feedback from humans. They assumed local linear relations between the robot motions and the ITD/ILD measures. This resulted in a controller which is simpler than the one presented in section 3.4. The scheme is similar to **Figure 3**, with an outer *vision* loop generating a reference motion, and *audio* modifying it.

4.3. Hybrid Control

Pomares et al. (2011) proposed a hybrid *vision/touch* controller for grasping objects, using a robot arm equipped with a hand. Visual feedback drives an active camera (installed on the robot tip) to observe the object and detect humans to be avoided, whereas touch feedback moves the fingers, to grasp the object. The authors defined matrix $\mathbf{S}$ in (7) to independently control arm and fingers with the respective sensor.

In Chatelain et al. (2017), a hybrid scheme controlled an ultrasonic probe in contact with the abdomen of a patient. The goal was to center the lesions in the ultrasound image observed by the surgeon. The probe was moved by projecting, via $\mathbf{S}$, the *touch*, and *vision* (from the ultrasound image) tasks in orthogonal directions.

4.4. Other Control Schemes

Some works do not strictly follow the classification given above. These are reviewed below.

The authors of Agravante et al. (2013, 2014) combined *vision* and *touch* to address joint human-humanoid table carrying. The table must stay flat, to prevent objects on top from falling off. Vision controlled the table inclination, whereas the forces exchanged with the human made the robot follow his/her intention. The approach is *shared*, with visual servoing in the outer loop of admittance control (**Figure 3**), to make all dof compliant. However, it is also *hybrid*, since some dof are controlled only with admittance. Specifically vision regulated only the table height in Agravante et al. (2013), and both table height and roll angle in Agravante et al. (2014).

The works (Cherubini and Chaumette, 2013; Cherubini et al., 2014) merged *vision* and *distance* to guarantee lidar-based obstacle avoidance during camera-based navigation. While following a pre-taught path, the robot must avoid obstacles which were not present before. Meanwhile, it moves the camera pan angle, to maintain scene visibility. Here, the selection matrix in (19) was a scalar function $\mathbf{S} \in [0, 1]$ dependent on the time-to-collision. In the safe context ($\mathbf{S} = 0$), the robot followed the taught path, with camera looking forward. In the unsafe context

($\mathbf{S} = 1$) the robot circumnavigated the obstacles. Therefore, the scheme is *hybrid* when $\mathbf{S} = 0$ or $\mathbf{S} = 1$ (i.e., vision and distance operate on independent components of the task vector), and *shared* when $\mathbf{S} \in (0, 1)$.

In Dean-Leon et al. (2016), proximity (*distance*) and tactile (*touch*) measurements controlled a robot arm in a pHRI scenario to avoid obstacles or—when contact is inevitable—to generate compliant behaviors. The framework linearly combined the two senses, and provided this signal to an inner admittance-like control loop (as in the *shared* scheme of **Figure 3**). Since the operation principle of both senses was complementary (one requires contact while the other does not), the integration can also be seen as *traded*.

The authors of Cherubini et al. (2015) enabled a robot to adapt to changes in the human behavior, during a human-robot collaborative screwing task. In contrast with classic hybrid vision–touch–position control, their scheme enabled smooth transitions, via weighted combinations of the tasks. The robot could execute *vision* and *force* tasks, either exclusively on different dof (*hybrid* approach) or simultaneously (*shared* approach).

5. CLASSIFICATION OF WORKS AND DISCUSSION

In this section, we use five criteria to classify all the surveyed papers which apply sensor-based control to collaborative robots. This taxonomy then serves as an inspiration to drive the following discussion on design choices, limitations, and future challenges.

In total, we refer to the 45 papers revised above. These include the works with only one sensor, discussed in section 3 (Maeda et al., 2001; Kumon et al., 2003, 2005; De Santis et al., 2007; Dune et al., 2008; Suphi Erden and Tomiyama, 2010; Suphi Erden and Maric, 2011; Tsui et al., 2011; Bussy et al., 2012; Flacco et al., 2012; Youssef et al., 2012; Ficuciello et al., 2013; Schlegl et al., 2013; Agustinos et al., 2014; Baumeyer et al., 2015; Gridseth et al., 2015, 2016; Magassouba et al., 2015, 2016a,b,c; Wang et al., 2015; Bauzano et al., 2016; Cai et al., 2016; Leboutet et al., 2016; Narayanan et al., 2016; Bergner et al., 2017; Cortesao and Dominici, 2017; Dean-Leon et al., 2017) and those which integrated multiple sensors, discussed in section 4 (Huang et al., 1999; Okuno et al., 2001, 2004; Natale et al., 2002; Hornstein et al., 2006; Pomares et al., 2011; Chan et al., 2012; Cherubini and Chaumette, 2013; Cherubini et al., 2014, 2015, 2016; Navarro et al., 2014; Papageorgiou et al., 2014; Dean-Leon et al., 2016; Chatelain et al., 2017). The five criteria are: sensor(s), integration method (when multiple sensors are used), control objective, target sector, and robot platform. In **Table 1**, we indicate these characteristics for each paper. Then, we focus on each characteristic, in **Tables 2**–**5**[4].

Table 2 classifies the papers according to the sensor/s. Column *mono* indicates the papers relying only on one sensor. For the others, we specify the integration approach (see section 4). Note that vision (alone or not) is by far the most popular sense, used in 22 papers. This comes as no surprise, since even for humans,

---

[4]In the Tables, we have used the following notation: V, T, A, D for Vision, Touch, Audition, and Distance, and sh., hyb., tra. for shared, hybrid, and traded.

TABLE 1 | Classification of all papers according to four criteria: sense(s) used by the robot, objective of the controller, target sector, and type of robot.

References	Sense(s)	Control objective	Sector	Robot
Cai et al. (2016) and Gridseth et al. (2016)	Vision	Contactless guidance	Service	Arm
Gridseth et al. (2015)	Vision	Remote guidance	Service	Arm
Dune et al. (2008), Tsui et al. (2011), and Narayanan et al. (2016)	Vision	Contactless guidance	Medical	Wheeled
Agustinos et al. (2014)	Vision	Contact w/humans	Medical	Arm
Bauzano et al. (2016)	Touch	Contact w/humans	Medical	Arm
		Remote guidance		
Cortesao and Dominici (2017)	Touch	Contact w/humans	Medical	Arm
Maeda et al. (2001), Suphi Erden and Tomiyama (2010), Suphi Erden and Maric (2011), and Ficuciello et al. (2013)	Touch	Direct guidance	Production	Arm
Wang et al. (2015)	Touch	Carrying	Production	Wheeled
Bussy et al. (2012)	Touch	Carrying	Production	Humanoid
Baumeyer et al. (2015)	Touch	Remote guidance	Medical	Arm
Kumon et al. (2003, 2005), Magassouba et al. (2016b)	Audition	Contactless guidance	Service	Heads
Magassouba et al. (2015, 2016a,c)	Audition	Contactless guidance	Service	Wheeled
De Santis et al. (2007), Flacco et al. (2012), and Schlegl et al. (2013)	Distance	Collision avoidance	Production	Arm
Leboutet et al. (2016), Bergner et al. (2017), and Dean-Leon et al. (2017)	Distance	Collision avoidance	Service	Arm
Cherubini et al. (2016)	V+T (tra.)	Assembly	Production	Arm
Okuno et al. (2001), Okuno et al. (2004), and Hornstein et al. (2006)	V+A(tra.)	Contactless guidance	Service	Heads
Chan et al. (2012)	V+A(tra.)	Contactless guidance	Service	Wheeled
Papageorgiou et al. (2014)	V+T+A+D (tra.)	Direct guidance	Medical	Wheeled
Navarro et al. (2014)	D+T(tra.)	Collision avoidance	Production	Arm
Huang et al. (1999)	D+A(tra.)	Collision avoidance	Service	Wheeled
Natale et al. (2002)	V+A(sh.)	Contactless guidance	Service	Heads
Pomares et al. (2011)	V+T(hyb.)	Collision avoidance	Production	Arm
Chatelain et al. (2017)	V+T (hyb.)	Contact w/humans	Medical	Arm
		Remote guidance		
Agravante et al. (2013, 2014)	V+T (sh.+hyb.)	Contact w/humans	Production	Humanoid
Cherubini and Chaumette (2013), Cherubini et al. (2014)	D+V (sh.+hyb.)	Collision avoidance	Production	Wheeled
Dean-Leon et al. (2016)	D+T (sh.+tra.)	Direct guidance	Service	Arm
Cherubini et al. (2015)	V+T (sh.+hyb.)	Assembly	Production	Arm

vision provides the richest perceptual information to structure the world and perform motion (Hoffman, 1998). Touch is the second most commonly used sensor (18 papers) and fundamental in pHRI, since it is the only one among the four that can be exploited directly to modulate contact.

Also note that, apart from Papageorgiou et al. (2014), no paper integrates more than two sensors. Given the sensors wide accessibility and the recent progress in computation power, this is probably due to the difficulty in designing a framework capable of managing such diverse and broad data. Another reason may be the presumed (but disputable) redundancy of the three contactless senses, which biases toward opting for vision, given its diffusion and popularity (also in terms of software). Touch—the only sensor measuring contact—is irreplaceable. This may also be the reason why, when merging two sensors, researchers have generally opted for vision+touch (7 out of 17 papers). The

most popular among the three integration methods is *traded control*, probably because it is the easiest to set up. In recent years, however, there has been a growing interest toward the *shared+hybrid* combination, which guarantees nice properties in terms of control smoothness.

An unexploited application of shared control is the combination of *vision* and *distance* (proximity sensors) to avoid collisions with humans. This can be formulated as in **Figure 3** by replacing touch control error $\mathbf{e_t}$ with an admittance-like distance control error:

$$\mathbf{e}_d = -(\mathbf{d} - \mathbf{d}^*) + \mathbf{M}(\ddot{\mathbf{x}} - \dot{\mathbf{x}}^r) + \mathbf{B}(\dot{\mathbf{x}} - \dot{\mathbf{x}}^r) + \mathbf{K}(\mathbf{x} - \mathbf{x}^r), \quad (20)$$

where $\mathbf{d}$ and $\mathbf{d}^*$ represent the measured and desired distance to obstacles. With this approach, the robot can stabilize at

TABLE 2 | Classification based on the sensors.

	Mono	Vision	Touch	Audition
Vision	Dune et al., 2008; Tsui et al., 2011; Agustinos et al., 2014; Gridseth et al., 2015, 2016; Cai et al., 2016; Narayanan et al., 2016			
Touch	Maeda et al., 2001; Suphi Erden and Tomiyama, 2010; Suphi Erden and Maric, 2011; Bussy et al., 2012; Ficuciello et al., 2013; Baumeyer et al., 2015; Wang et al., 2015; Bauzano et al., 2016; Cortesao and Dominici, 2017	tra. (Cherubini et al., 2016), hyb. (Pomares et al., 2011; Chatelain et al., 2017) sh.+hyb. (Agravante et al., 2013, 2014; Cherubini et al., 2015)		
Audition	Kumon et al. (2003, 2005), Youssef et al. (2012), Magassouba et al. (2015, 2016a,b,c)	tra. Okuno et al (2001, 2004), Hornstein et al. (2006), Chan et al. (2012), Papageorgiou et al. (2014), sh. (Natale et al., 2002)	tra. (Papageorgiou et al., 2014)	
Distance	De Santis et al., 2007; Flacco et al., 2012; Schlegl et al., 2013; Leboutet et al., 2016; Bergner et al., 2017; Dean-Leon et al., 2017	sh.+hyb. (Cherubini and Chaumette, 2013; Cherubini et al., 2014)	sh.+tra. (Dean-Leon et al., 2016) tra. (Navarro et al., 2014)	tra. (Huang et al., 1999; Papageorgiou et al., 2014)

a given "safe" distance from an obstacle, or move away from it.

In the authors' opinion, no sensor(s) nor (if needed) integration method is the best, and the designer should choose according to the objective at stake. For this, nature and evolution can be extremely inspiring but technological constraints (e.g., hardware and software availability) must also be accounted for, with the golden rule of engineering that "simpler is better."

Table 3 classifies the papers according to the *control objective*. In the table, we also apply the taxonomy of pHRI layers introduced in De Luca and Flacco (2012), and evoked in the introduction: *safety*, *coexistence*, *collaboration*. Works that focus on collision avoidance address *safety*, and works where the robot acts on passive humans address *coexistence*. For the *collaboration* layer, we distinguish two main classes of works. First, those where the human was guiding the robot (without contact, with direct contact, or with remote physical contact as in tele-operation), then those where the two collaborated (e.g., for part assembly or object carrying). The idea (also in line with De Luca and Flacco, 2012) is the lower lines in the table generally require higher cognitive capabilities (e.g., better modeling of environment and task). Some works, particularly in the field of medical robotics (Agustinos et al., 2014; Bauzano et al., 2016; Chatelain et al., 2017) cover both coexistence and collaboration, since the human guided the robot to operate on another human. Interestingly, the senses appear in the table with a trend analogous to biology. *Distance* is fundamental for collision avoidance, when the human is far, and his/her role in the interaction is basic (s/he is mainly perceived as an obstacle). Then, audio is used for contactless guidance. As human and robot are closer, *touch* takes over the role of *audio*. As mentioned above, *vision* is a transversal sense, capable of covering most distance ranges. Yet, when contact is present (i.e., in the four lower lines), it is systematically complemented by touch, a popular pairing as also shown in **Table 2** and discussed above.

TABLE 3 | Classification based on the control objective with corresponding pHRI layer as proposed in De Luca and Flacco (2012) (in parenthesis).

Collision avoidance (*safety*)	Distance (De Santis et al., 2007; Flacco et al., 2012; Schlegl et al., 2013; Leboutet et al., 2016; Bergner et al., 2017; Dean-Leon et al., 2017), distance+touch (Navarro et al., 2014),
	Distance+audition (Huang et al., 1999), vision+touch (Pomares et al., 2011),
	Vision+distance (Cherubini and Chaumette, 2013; Cherubini et al., 2014)
Contact with passive humans (*coexistence*)	Vision (Agustinos et al., 2014), touch (Bauzano et al., 2016; Cortesao and Dominici, 2017),
	Vision+touch (Chatelain et al., 2017)
Contactless guidance (*collaboration*)	Vision (Dune et al., 2008; Tsui et al., 2011; Cai et al., 2016; Gridseth et al., 2016; Narayanan et al., 2016)
	Audition (Kumon et al., 2005; Youssef et al., 2012; Magassouba et al., 2015, 2016a,b,c)
	Vision+audition (Okuno et al., 2001, 2004; Natale et al., 2002; Hornstein et al., 2006; Chan et al., 2012)
Direct guidance (*collaboration*)	Touch+audition+distance+vision (Papageorgiou et al., 2014),
	Touch (Maeda et al., 2001; Suphi Erden and Tomiyama, 2010; Suphi Erden and Maric, 2011; Ficuciello et al., 2013), touch+distance (Dean-Leon et al., 2016)
Remote guidance (*collaboration*)	Vision (Agustinos et al., 2014; Gridseth et al., 2015), touch (Baumeyer et al., 2015; Bauzano et al., 2016),
	Vision+touch (Chatelain et al., 2017)
Collaborative assembly (*collaboration*)	Vision+touch (Cherubini et al., 2015, 2016)
Collaborative carrying (*collaboration*)	Touch (Bussy et al., 2012; Wang et al., 2015), vision+touch (Agravante et al., 2013, 2014)

Table 4 classifies the papers according to the target (or *potential*) sector. We propose three sectors: *Production*, *Medical*, and *Service*. Production is the historical sector of robotics; applications include: manufacturing (assembly, welding,

TABLE 4 | Classification based on target/potential sectors.

Production (manufacturing, transportation, construction)	Touch (Maeda et al., 2001; Suphi Erden and Tomiyama, 2010; Suphi Erden and Maric, 2011; Bussy et al., 2012; Ficuciello et al., 2013; Wang et al., 2015), distance (De Santis et al., 2007; Flacco et al., 2012; Schlegl et al., 2013),
	D+T (Navarro et al., 2014) V+T (Pomares et al., 2011; Agravante et al., 2013, 2014; Cherubini et al., 2015, 2016),
	V+D (Cherubini and Chaumette, 2013; Cherubini et al., 2014)
Medical (surgery, diagnosis, assistance)	Vision (Dune et al., 2008; Tsui et al., 2011; Agustinos et al., 2014; Narayanan et al., 2016), touch (Baumeyer et al., 2015; Bauzano et al., 2016; Cortesao and Dominici, 2017),
	V+T+A+D (Papageorgiou et al., 2014), V+T (Chatelain et al., 2017)
Service (companionship, domestic, personal)	Vision (Gridseth et al., 2015, 2016; Cai et al., 2016), audition (Kumon et al., 2005; Youssef et al., 2012; Magassouba et al., 2015, 2016a,b,c),
	distance (Leboutet et al., 2016; Bergner et al., 2017; Dean-Leon et al., 2017), V+A (Okuno et al., 2001, 2004; Natale et al., 2002; Hornstein et al., 2006; Chan et al., 2012),
	D+A (Huang et al., 1999), T+D (Dean-Leon et al., 2016)

TABLE 5 | Classification based on the type of robot platform.

Arms	Vision (Agustinos et al., 2014; Gridseth et al., 2015, 2016; Cai et al., 2016), touch (Maeda et al., 2001; Suphi Erden and Tomiyama, 2010; Suphi Erden and Maric, 2011; Ficuciello et al., 2013; Baumeyer et al., 2015; Bauzano et al., 2016; Cortesao and Dominici, 2017), distance (De Santis et al., 2007; Flacco et al., 2012; Schlegl et al., 2013; Leboutet et al., 2016; Bergner et al., 2017; Dean-Leon et al., 2017),
	V+T (Pomares et al., 2011; Cherubini et al., 2015, 2016; Chatelain et al., 2017), D+T (Navarro et al., 2014; Dean-Leon et al., 2016)
Wheeled	Vision (Dune et al., 2008; Tsui et al., 2011; Narayanan et al., 2016), touch (Wang et al., 2015), audition (Magassouba et al., 2015, 2016a,b), V+A (Chan et al., 2012), V+T+A+D (Papageorgiou et al., 2014), D+A (Huang et al., 1999), V+D (Cherubini and Chaumette, 2013; Cherubini et al., 2014)
Humanoids	Touch (Bussy et al., 2012), V+T (Agravante et al., 2013, 2014)
Heads	Audition (Kumon et al., 2003, 2005; Magassouba et al., 2016b), V+A (Okuno et al., 2001, 2004; Natale et al., 2002; Hornstein et al., 2006)

pick-and-place), transportation (autonomous guided vehicles, logistics) and construction (material and brick transfer). The medical category has become very popular in recent years, with applications spanning from robotic surgery (surgical grippers and needle manipulation), diagnosis (positioning of ultrasonic probes; Tirindelli et al., 2020 or endoscopes), and assistance (intelligent wheelchairs, feeding and walking aids). The service sector is the one that in the authors' opinion presents the highest potential for growth in the coming years. Applications include companionship (elderly and child care), domestic (cleaning, object retrieving), personal (chat partners, tele-presence). The table shows that all four sensors have been deployed in all three sectors. The only exception is *audition* not being used in *production* applications, probably because of the noise—common in industrial environments.

Finally, **Table 5** gives a classification based on the robotic platform. We can see that (unsurprisingly) most works use fixed base *arms*. The second most used platforms here are *wheeled* robots. Then, the *humanoids* category, which refers to robots with anthropomorphic design (two arms and biped locomotion capabilities). Finally, we consider robot *heads*, which are used exclusively for audio-based control. While robot heads are commonly used for face tracking in *Social Human Robot Interaction*, such works are not reviewed in this survey as they do not generally involve contact.

6. CONCLUSIONS

This work presents a systematic review of sensor-based controllers which enable collaboration and/or interaction between humans and robots. We considered four senses:

vision, touch, audition, and distance. First, we introduce a tutorial-like general formulation of sensor-based control (Navarro-Alarcon et al., 2020), which we instantiate for visual servoing, touch control, aural servoing, and distance-based control, while reviewing representative papers. Next, with the same formulation, we model the methods that integrate multiple sensors, while again discussing related works. Finally, we classify the surveyed body of literature according to: used sense(s), integration method, control objective, target application, and platform.

Although vision and touch (*proprioceptive force* rather than *tact*) emerge nowadays as the most popular senses on collaborative robots, the advent of cheap, precise, and easy to integrate tactile, distance, and audio sensors present great opportunities for the future. Typically, we believe that robot skins (e.g., on arms and hands, Guadarrama-Olvera et al., 2019; Navarro et al., 2020) will simplify interaction, boosting the opportunities for human-robot collaboration. It is imperative that researchers develop the appropriate tools for this. Distance/proximity feedback is promising to fully perceive the human operating near the robot (something monocular vision cannot do). Audio feedback is key for developing robotic heads that can interact in a natural way with human speakers.

Finally, some open problems must be addressed, to develop robust controllers for real-world applications. For example, the use of task constraints has not been sufficiently explored when multiple sensors are integrated. Also, difficulty in obtaining models describing and predicting human behavior hampers the implementation of human-robot collaborative tasks. The use of multimodal data such as RGB-D cameras with multiple proximity sensors may be an interesting solution for this human motion sensing and estimation problem. More research needs to be conducted in this direction.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/supplementary materials, further inquiries can be directed to the corresponding author/s.

AUTHOR CONTRIBUTIONS

FUNDING

REFERENCES

Abderrahmane, Z., Ganesh, G., Crosnier, A., and Cherubini, A. (2018). Haptic zero-shot learning: recognition of objects never touched before. *Robot. Auton. Syst.* 105, 11–25. doi: 10.1016/j.robot.2018.03.002

Agravante, D. J., Cherubini, A., Bussy, A., Gergondet, P., and Kheddar, A. (2014). "Collaborative human-humanoid carrying using vision and haptic sensing," in *IEEE Int. Conf. on Robotics and Automation, ICRA.* doi: 10.1109/ICRA.2014.6906917

Agravante, D. J., Cherubini, A., Bussy, A., and Kheddar, A. (2013). "Human-humanoid joint haptic table carrying task with height stabilization using vision," in *IEEE/RSJ Int. Conf. on Robots and Intelligent Systems, IROS.* doi: 10.1109/IROS.2013.6697019

Agustinos, A., Wolf, R., Long, J. A., Cinquin, P., and Voros, S. (2014). "Visual servoing of a robotic endoscope holder based on surgical instrument tracking," in *IEEE RAS/EMBS Int. Conf. on Biomedical Robotics and Biomechatronics*, 13–18. doi: 10.1109/BIOROB.2014.6913744

Ajoudani, A., Zanchettin, A. M., Ivaldi, S., Albu-Schäffer, A., Kosuge, K., and Khatib, O. (2017). Progress and prospects of the human-robot collaboration. *Auton. Robots* 42, 957–975. doi: 10.1007/s10514-017-9677-2

Argall, B. D., and Billard, A. G. (2010). A survey of tactile human-robot interactions. *Robot. Auton. Syst.* 58, 1159–1176. doi: 10.1016/j.robot.2010.07.002

Azizian, M., Khoshnam, M., Najmaei, N., and Patel, R. V. (2014). Visual Servoing in medical robotics: a survey. Part I: endoscopic and direct vision imaging-techniques and applications. *Int. J. Med. Robot.* 10, 263–274. doi: 10.1002/rcs.1531

Baeten, J., Bruyninckx, H., and De Schutter, J. (2003). Integrated vision/force robotic servoing in the task frame formalism. *Int. J. Robot. Res.* 22, 941–954. doi: 10.1177/027836490302210010

Baumeyer, J., Vieyres, P., Miossec, S., Novales, C., Poisson, G., and Pinault, S. (2015). "Robotic co-manipulation with 6 DOF admittance control: application to patient positioning in proton-therapy," in *IEEE Int. Work. on Advanced Robotics and its Social Impacts*, 1–6. doi: 10.1109/ARSO.2015.7428220

Bauzano, E., Estebanez, B., Garcia-Morales, I., and Munoz, V. F. (2016). Collaborative human-robot system for HALS suture procedures. *IEEE Syst. J.* 10, 957–966. doi: 10.1109/JSYST.2014.2299559

Bergner, F., Dean-Leon, E., and Cheng, G. (2017). "Efficient event-driven reactive control for large scale robot skin," in *IEEE Int. Conf. on Robotics and Automation, ICRA*, 394–400. doi: 10.1109/ICRA.2017.7989051

Berthoz, A. (2002). *The Brain's Sense of Movement.* Harvard University Press.

Bicchi, A., Peshkin, M., and Colgate, J. (2008). *Safety for Physical Human-Robot Interaction.* Springer Handbook of Robotics. doi: 10.1007/978-3-540-30301-5_58

Bussy, A., Kheddar, A., Crosnier, A., and Keith, F. (2012). "Human-humanoid haptic joint object transportation case study," in *IEEE/RSJ Int. Conf. on Robots and Intelligent Systems, IROS*, 3633–3638. doi: 10.1109/IROS.2012.6385921

Cai, C., Somani, N., and Knoll, A. (2016). Orthogonal image features for visual servoing of a 6-dof manipulator with uncalibrated stereo cameras. *IEEE Trans. Robot.* 32, 452–461. doi: 10.1109/TRO.2016.2535443

Chan, V., Jin, C., and van Schaik, A. (2012). Neuromorphic audio-visual sensor fusion on a sound-localising robot. *Front. Neurosci.* 6:21. doi: 10.3389/fnins.2012.00021

Chatelain, P., Krupa, A., and Navab, N. (2017). Confidence-driven control of an ultrasound probe. *IEEE Trans. Robot.* 33, 1410–1424. doi: 10.1109/TRO.2017.2723618

Chaumette, F., and Hutchinson, S. (2006). Visual servo control, Part I: basic approaches. *IEEE Robot. Autom. Mag.* 13, 82–90. doi: 10.1109/MRA.2006.250573

Cherubini, A., and Chaumette, F. (2013). Visual navigation of a mobile robot with laser-based collision avoidance. *Int. J. Robot. Res.* 32, 189–209. doi: 10.1177/0278364912460413

Cherubini, A., Passama, R., Crosnier, A., Lasnier, A., and Fraisse, P. (2016). Collaborative manufacturing with physical human-robot interaction. *Robot. Comput. Integr. Manufact.* 40, 1–13. doi: 10.1016/j.rcim.2015.12.007

Cherubini, A., Passama, R., Fraisse, P., and Crosnier, A. (2015). A unified multimodal control framework for human-robot interaction. *Robot. Auton. Syst.* 70, 106–115. doi: 10.1016/j.robot.2015.03.002

Cherubini, A., Spindler, F., and Chaumette, F. (2014). Autonomous visual navigation and laser-based moving obstacle avoidance. *IEEE Trans. Int. Transport. Syst.* 15, 2101–2110. doi: 10.1109/TITS.2014.2308977

Colgate, J., Wannasuphoprasit, W., and Peshkin, M. (1996). "Cobots: robots for collaboration with human operators," in *Proc ASME Dynamic Systems and Control Division, Vol. 58*, 433–439.

Cortesao, R., and Dominici, M. (2017). Robot force control on a beating heart. *IEEE/ASME Trans. Mechatron.* 22, 1736–1743. doi: 10.1109/TMECH.2017.2696259

Davison, E., and Goldenberg, A. (1975). Robust control of a general servomechanism problem: the servo compensator. *IFAC Proc.* 8(1, Part 1), 231–239. doi: 10.1016/S1474-6670(17)67744-9

De Luca, A., and Flacco, F. (2012). "Integrated control for pHRI: collision avoidance, detection, reaction and collaboration," in *IEEE RAS/EMBS Int. Conf. on Biomedical Robotics and Biomechatronics, BIOROB.* doi: 10.1109/BioRob.2012.6290917

De Santis, A., Lippiello, V., Siciliano, B., and Villani, L. (2007). Human-robot interaction control using force and vision. *Adv. Control Theor. Appl.* 353, 51–70. doi: 10.1007/978-3-540-70701-1_3

Dean-Leon, E., Bergner, F., Ramirez-Amaro, K., and Cheng, G. (2016). "From multi-modal tactile signals to a compliant control," in *IEEE-RAS Int. Conf. on Humanoid Robots*, 892–898. doi: 10.1109/HUMANOIDS.2016.7803378

Dean-Leon, E., Pierce, B., Bergner, F., Mittendorfer, P., Ramirez-Amaro, K., Burger, W., et al. (2017). "TOMM: Tactile omnidirectional mobile manipulator," in *IEEE Int. Conf. on Robotics and Automation, ICRA*, 2441–2447. doi: 10.1109/ICRA.2017.7989284

Del Prete, A., Denei, S., Natale, L., Mastrogiovanni, F., Nori, F., Cannata, G., et al. (2011). "Skin spatial calibration using force/torque measurements," in *IEEE/RSJ Int. Conf. on Robots and Intelligent Systems.* doi: 10.1109/IROS.2011.6094896

Dune, C., Remazeilles, A., Marchand, E., and Leroux, C. (2008). "Vision-based grasping of unknown objects to improve disabled people autonomy," in *Robotics: Science and Systems.*

Ernst, M. O., and Banks, M. S. (2002). Humans integrate visual and haptic information in a statistically optimal fashion. *Nature* 415, 429–433. doi: 10.1038/415429a

Ficuciello, F., Romano, A., Villani, L., and Siciliano, B. (2013). "Cartesian impedance control of redundant manipulators for human-robot

co-manipulation," in *IEEE/RSJ Int. Conf. on Robots and Intelligent Systems, IROS*. doi: 10.1109/IROS.2014.6942847

Flacco, F., Kroger, T., De Luca, A., and Khatib, O. (2012). "A depth space approach to human-robot collision avoidance," in *IEEE Int. Conf. on Robotics and Automation, ICRA*. doi: 10.1109/ICRA.2012.6225245

Gao, B., Li, H., Li, W., and Sun, F. (2016). 3D moth-inspired chemical plume tracking and adaptive step control strategy. *Adapt. Behav.* 24, 52–65. doi: 10.1177/1059712315623998

Göger, D., Blankertz, M., and Wörn, H. (2010). A tactile proximity sensor. *IEEE Sensors* 589–594.

Gridseth, M., Hertkorn, K., and Jagersand, M. (2015). "On visual servoing to improve performance of robotic grasping," in *Conf. on Computer and Robot Vision*, 245–252. doi: 10.1109/CRV.2015.39

Gridseth, M., Ramirez, O., Quintero, C. P., and Jagersand, M. (2016). "Vita: Visual task specification interface for manipulation with uncalibrated visual servoing," in *IEEE Int. Conf. on Robotics and Automation, ICRA*. doi: 10.1109/ICRA.2016.7487521

Guadarrama-Olvera, J. R., Dean-Leon, E., Bergner, F., and Cheng, G. (2019). Pressure-driven body compliance using robot skin. *IEEE Robot. Autom. Lett.* 4, 4418–4423. doi: 10.1109/LRA.2019.2928214

Ha, D., Sun, Q., Su, K., Wan, H., Li, H., Xu, N., et al. (2015). Recent achievements in electronic tongue and bioelectronic tongue as taste sensors. *Sensors Actuat. Part B* 207, 1136–1146. doi: 10.1016/j.snb.2014.09.077

Haddadin, S., De Luca, A., and Albu-Schäffer, A. (2017). Robot collisions: a survey on detection, isolation, and identification. *IEEE Trans. Robot.* 33, 1292–1312. doi: 10.1109/TRO.2017.2723903

Hoffman, D. (1998). *Visual Intelligence: How We Create what We See*. W. W. Norton and Company.

Hogan, N. (1985). Impedance control: an approach to manipulation: parts I-III. *ASME J. Dyn. Syst. Measure. Control* 107, 1–24. doi: 10.1115/1.3140701

Hornstein, J., Lopes, M., Santos-Victor, J., and Lacerda, F. (2006). "Sound localization for humanoid robots - building audio-motor maps based on the HRTF," in *IEEE/RSJ Int. Conf. on Robots and Intelligent Systems, IROS*, 1170–1176. doi: 10.1109/IROS.2006.281849

Huang, J., Supaongprapa, T., Terakura, I., Wang, F., Ohnishi, N., and Sugie, N. (1999). A model-based sound localization system and its application to robot navigation. *Robot. Auton. Syst.* 27, 199–209. doi: 10.1016/S0921-8890(99)00002-0

ISO 13482:2014 (2014). *Robots and Robotic Devices - Safety Requirements for Personal Care Robots*. Technical report, International Organization for Standardization, Geneva.

Khatib, O. (1985). "Real-time obstacle avoidance for manipulators and mobile robots," in *IEEE Int. Conf. on Robotics and Automation, ICRA*. doi: 10.1109/ROBOT.1985.1087247

Kobayashi, Y., Habara, M., Ikezazki, H., Chen, R., Naito, Y., and Toko, K. (2010). Advanced taste sensors based on artificial lipids with global selectivity to basic taste qualities and high correlation to sensory scores. *Sensors* 10, 3411–3443. doi: 10.3390/s100403411

Kowadlo, G., and Russell, R. A. (2008). Robot odor localization: a taxonomy and survey. *Int. J. Robot. Res.* 27, 869–894. doi: 10.1177/0278364908095118

Kumon, M., Shimoda, T., Kohzawa, R., Mizumoto, I., and Iwai, Z. (2005). "Audio servo for robotic systems with pinnae," in *IEEE/RSJ Int. Conf. on Robots and Intelligent Systems, IROS*, 1881–1886. doi: 10.1109/IROS.2005.1545092

Kumon, M., Sugawara, T., Miike, K., Mizumoto, I., and Iwai, Z. (2003). "Adaptive audio servo for multirate robot systems," in *IEEE/RSJ Int. Conf. on Robots and Intelligent Systems, IROS, Vol. 1*, 182–187. doi: 10.1109/IROS.2003.1250625

La Valle, S. M. (2006). *Planning Algorithms*. Cambridge University Press.

Leboutet, Q., Dean-León, E., and Cheng, G. (2016). "Tactile-based compliance with hierarchical force propagation for omnidirectional mobile manipulators," in *IEEE-RAS Int. Conf. on Humanoid Robots*. doi: 10.1109/HUMANOIDS.2016.7803383

Li, Q., Schürman, C., Haschke, R., and Ritter, H. (2013). "A control framework for tactile servoing," in *Robotics: Science and Systems (RSS)*. doi: 10.15607/RSS.2013.IX.045

Lin, C. H., Fishel, J. A., and Loeb, G. E. (2013). *Estimating Point of Contact, Force and Torque in a Biomimetic Tactile Sensor With Deformable Skin*. Technical report, SynTouch LLC.

Maeda, Y., Hara, T., and Arai, T. (2001). "Human-robot cooperative manipulation with motion estimation," in *IEEE/RSJ Int. Conf. on Robots and Intelligent Systems, IROS, Vol. 4*, 2240–2245. doi: 10.1109/IROS.2001.976403

Magassouba, A., Bertin, N., and Chaumette, F. (2015). "Sound-based control with two microphones," in *IEEE/RSJ Int. Conf. on Robots and Intelligent Systems, IROS*, 5568–5573. doi: 10.1109/IROS.2015.7354166

Magassouba, A., Bertin, N., and Chaumette, F. (2016a). "Audio-based robot control from interchannel level difference and absolute sound energy," in *IEEE/RSJ Int. Conf. on Robots and Intelligent Systems, IROS*, 1992–1999. doi: 10.1109/IROS.2016.7759314

Magassouba, A., Bertin, N., and Chaumette, F. (2016b). "Binaural auditory interaction without HRTF for humanoid robots: a sensor-based control approach," in *See, Touch, and Hear: 2nd Workshop on Multimodal Sensor-based Robot Control for HRI and Soft Manipulation, IROS*.

Magassouba, A., Bertin, N., and Chaumette, F. (2016c). "First applications of sound-based control on a mobile robot equipped with two microphones," in *IEEE Int. Conf. on Robotics and Automation, ICRA*, 2557–2562. doi: 10.1109/ICRA.2016.7487411

Morel, G., Malis, E., and Boudet, S. (1998). "Impedance based combination of visual and force control," in *IEEE Int. Conf. on Robotics and Automation, ICRA, Vol. 2*, 1743–1748. doi: 10.1109/ROBOT.1998.677418

Nakadai, K., Nakajima, H., Murase, M., Kaijiri, S., Yamada, K., Nakamura, T., et al. (2006). "Robust tracking of multiple sound sources by spatial integration of room and robot microphone arrays," in *IEEE Int. Conf. on Acoustics Speech and Signal Processing*. doi: 10.1109/ICASSP.2006.1661122

Narayanan, K. V., Pasteau, F., Marchal, M., Krupa, A., and Babel, M. (2016). Vision-based adaptive assistance and haptic guidance for safe wheelchair corridor following. *Comput. Vis. Image Underst.* 149, 171–185. doi: 10.1016/j.cviu.2016.02.008

Natale, L., Metta, G., and Sandini, G. (2002). Development of auditory-evoked reflexes: visuo-acoustic cues integration in a binocular head. *Robot. Auton. Syst.* 39, 87–106. doi: 10.1016/S0921-8890(02)00174-4

Natale, L., and Torres-Jara, E. (2006). "A sensitive approach to grasping," in *Proc. of the 6th Int. Workshop on Epigenetic Robotics*.

Navarro, S. E., Nagels, S., Alagi, H., Faller, L., Goury, O., Morales-Bieze, T., et al. (2020). A model-based sensor fusion approach for force and shape estimation in soft robotics. *IEEE Robot. Autom. Lett.* 5, 5621–5628. doi: 10.1109/LRA.2020.3008120

Navarro, S. E., Schonert, M., Hein, B., and WWörn, H. (2014). "6D proximity servoing for preshaping and haptic exploration using capacitive tactile proximity sensors," in *IEEE/RSJ Int. Conf. on Robots and Intelligent Systems, IROS*.

Navarro-Alarcon, D., Qi, J., Zhu, J., and Cherubini, A. (2020). A Lyapunov-stable adaptive method to approximate sensorimotor models for sensor-based control. *Front. Neurorobot.* 14:59. doi: 10.3389/fnbot.2020.00059

Nelson, B. J., and Khosla, P. K. (1996). Force and vision resolvability for assimilating disparate sensory feedback. *IEEE Trans. Robot. Autom.* 12, 714–731. doi: 10.1109/70.538976

Nelson, B. J., Morrow, J. D., and Khosla, P. K. (1995). "Improved force control through visual servoing," in *Proc. of the American Control Conference, Vol. 1*, 380–386. doi: 10.1109/ACC.1995.529274

Nocedal, J., and Wright, S. (2000). *Numerical Optimization*. Springer Series in Operations Research and Financial Engineering. doi: 10.1007/b98874

Okuno, H. G., Nakadai, K., Hidai, K. I., Mizoguchi, H., and Kitano, H. (2001). "Human-robot interaction through real-time auditory and visual multiple-talker tracking," in IEEE/RSJ Int. Conf. on Robots and Intelligent Systems, IROS, Vol. 3, 1402–1409. doi: 10.1109/IROS.2001.977177

Okuno, H. G., Nakadai, K., Lourens, T., and Kitano, H. (2004). Sound and visual tracking for humanoid robot. *Appl. Intell.* 20, 253–266. doi: 10.1023/B:APIN.0000021417.62541.e0

Papageorgiou, X. S., Tzafestas, C. S., Maragos, P., Pavlakos, G., Chalvatzaki, G., Moustris, G., et al. (2014). "Advances in intelligent mobility assistance robot integrating multimodal sensory processing," in *Universal Access in Human-Computer Interaction. Aging and Assistive Environments* (Springer International Publishing), 692–703. doi: 10.1007/978-3-319-07446-7_66

Passenberg, C., Peer, A., and Buss, M. (2010). A survey of environment- operator- and task-adapted controllers for teleoperation systems. *Mechatronics* 20, 787–801. doi: 10.1016/j.mechatronics.2010.04.005

Phoha, S. (2014). Machine perception and learning grand challenge: situational intelligence using cross-sensory fusion. *Front. Robot.* AI 1:7. doi: 10.3389/frobt.2014.00007

Pomares, J., Perea, I., Garcia, G. J., Jara, C. A., Corrales, J. A., and Torres, F. (2011). A multi-sensorial hybrid control for robotic manipulation in human-robot workspaces. *Sensors* 11, 9839–9862. doi: 10.3390/s111009839

Proske, U., and Gandevia, S. C. (2012). The proprioceptive senses: their roles in signaling body shape, body position and movement, and muscle force. *Physiol. Rev.* 92, 1651–1697. doi: 10.1152/physrev.00048.2011

Rahbar, F., Marjovi, A., Kibleur, P., and Martinoli, A. (2017). "A 3-D bio-inspired odor source localization and its validation in realistic environmental conditions," in *IEEE/RSJ Int. Conf. on Robots and Intelligent Systems, IROS*, 3983–3989. doi: 10.1109/IROS.2017.8206252

Raibert, M. H., and Craig, J. J. (1981). Hybrid position/force control of manipulators. *ASME J. Dyn. Syst. Meas. Control* 126–133. doi: 10.1115/1.3139652

Rayleigh, L. (1907). On our perception of sound direction. *Lond. Edinburgh Dublin Philos. Mag. J. Sci.* 13, 214–232. doi: 10.1080/14786440709463595

Russell, R. A. (2006). "Tracking chemical plumes in 3-dimensions," in *IEEE Int. Conf. on Robotics and Biomimetics*, 31–36. doi: 10.1109/ROBIO.2006.340274

Schlegl, T., Kröger, T., Gaschler, A., Khatib, O., and Zangl, H. (2013). "Virtual whiskers-highly responsive robot collision avoidance," in *IEEE/RSJ Int. Conf. on Robots and Intelligent Systems*, IROS. doi: 10.1109/IROS.2013.6697134

Schmitz, A., Maiolino, P., Maggiali, M., Natale, L., Cannata, G., and Metta, G. (2011). Methods and technologies for the implementation of large-scale robot tactile sensors. *IEEE Trans. Robot.* 27, 389–400. doi: 10.1109/TRO.2011.2132930

Shimazu, H., Kobayashi, K., Hashimoto, A., and Kameoka, T. (2007). "Tasting robot with an optical tongue: real time examining and advice giving on food and drink," in *Human Interface and the Management of Information. Methods, Techniques and Tools in Information Design*, eds M. J. Smith and G. Salvendy (Berlin; Heidelberg: Springer). doi: 10.1007/978-3-540-73345-4_107

Smits, R., De Laet, T., Claes, K., Bruyninckx, H., and De Schutter, J. (2008). "iTASC: a tool for multi-sensor integration in robot manipulation," in *IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, 426–433. doi: 10.1109/MFI.2008.4648032

Suphi Erden, M., and Maric, B. (2011). Assisting manual welding with robot. *Robot. Comput. Integr. Manufact.* 27, 818–828. doi: 10.1016/j.rcim.2011.01.003

Suphi Erden, M., and Tomiyama, T. (2010). Human intent detection and physically interactive control of a robot without force sensors. *IEEE Trans. Robot.* 26, 370–382. doi: 10.1109/TRO.2010.2040202

Taylor, G., and Kleeman, L. (2006). *Visual Perception and Robotic Manipulation: 3D Object Recognition, Tracking and Hand-Eye Coordination*. Springer Tracts in Advanced Robotics. Springer. doi: 10.1007/978-3-540-3 3455-2

Tirindelli, M., Victorova, M., Esteban, J., Kim, S. T., Navarro-Alarcon, D., Zheng, Y. P., et al. (2020). Force-ultrasound fusion: Bringing spine robotic-us to the next "level". *IEEE Robot. Autom. Lett.* 5, 5661–5668. doi: 10.1109/LRA.2020.3009069

Tsui, K. M., Kim, D.-J., Behal, A., Kontak, D., and Yanco, H. A. (2011). I want that: Human-in-the-loop control of a wheelchair-mounted robotic arm. *Appl. Bionics Biomech.* 8, 127–147. doi: 10.1155/2011/698079

Villani, L., and De Schutter, J. (2008). "Chapter 7: Force control," in *Springer Handbook of Robotics*, eds B. Siciliano and O. Khatib (Springer), 161–185. doi: 10.1007/978-3-540-30301-5_8

Villani, V., Pini, F., Leali, F., and Secchi, C. (2018). Survey on human-robot collaboration in industrial settings: safety, intuitive interfaces and applications. *Mechatronics* 55, 248–266. doi: 10.1016/j.mechatronics.2018.02.009

Wang, Y., Smith, C., Karayiannidis, Y., and Ögren, P. (2015). "Cooperative control of a serial-to-parallel structure using a virtual kinematic chain in a mobile dual-arm manipulation application," in *IEEE/RSJ Int. Conf. on Robots and Intelligent Systems, IROS*, 2372–2379. doi: 10.1109/IROS.2015.7353698

Wettels, N., Santos, V. J., Johansson, R. S., and Loeb, G. (2008). Biomimetic tactile sensor array. *Adv. Robot.* 22, 829–849. doi: 10.1163/156855308X314533

Whitney, D. (1969). Resolved motion rate control of manipulators and human prostheses. *IEEE Trans. Man-Mach. Syst.* 10, 47–53. doi: 10.1109/TMMS.1969.299896

Youssef, K., Argentieri, S., and Zarader, J. L. (2012). "Towards a systematic study of binaural cues," in *IEEE/RSJ Int. Conf. on Robots and Intelligent Systems, IROS*, 1004–1009. doi: 10.1109/IROS.2012.6385554

Zhang, H., and Chen, N. N. (2000). Control of contact via tactile sensing. *IEEE Trans. Robot. Autom.* 16, 482–495. doi: 10.1109/70.880799

Check for
updates

Binary and Hybrid Work-Condition Maps for Interactive Exploration of Ergonomic Human Arm Postures

Luka Peternel[1]*, Daniel Tofte Schøn[2] and Cheng Fang[2]

[1] Delft Haptics Lab, Department of Cognitive Robotics, Delft University of Technology, Delft, Netherlands, [2] SDU Robotics, The Maersk Mc-Kinney Moller Institute, University of Southern Denmark, Odense, Denmark

Ergonomics of human workers is one of the key elements in design and evaluation of production processes. Human ergonomics have a major impact on productivity as well as chronic health risks incurred by inappropriate working postures and conditions. In this paper we propose a novel method for estimating and communicating the ergonomic work condition called *Binary Work-Condition Map*, which provides a visualized feedback about work conditions of different configurations of an arm. The map is of binary nature and is derived by imposing the desired thresholds on considered ergonomic and safety related criteria. Therefore, the suggested arm postures in the map guarantee that all considered criteria are satisfied. This eliminates the ambiguity compared to state-of-the-art maps that uses continuous scales derived from weighted sum of multiple ergonomics criteria. In addition, to combine the advantages of both the binary map and the continuous map, we additionally propose a *Hybrid Work-Condition Map* that rules out unsuitable workspace with the binary map approach and renders the suitable workspace with the continuous map approach. The proposed approach was tested in simulation for various tasks and conditions. In addition, we conducted subjective evaluation experiments to compare the proposed methods with the state-of-the art method regarding the usability. The results indicated that the binary map is simpler to use, while the hybrid map is a good tradeoff between the binary and the continuous map. In selecting the map, strong points of each map should be considered with respect to the requirements of a specific application and task.

Keywords: Work-Condition Map, ergonomic human arm posture, biomechanical model, graphical user interface, interactive exploration

1. INTRODUCTION

Robots have successfully supplemented human workers in modern manufacturing processes. Nevertheless, in many cases, robots did not replace the human workers, who are still an essential element at various production stages. While robots can work safely and efficiently without getting tired for extended periods of time, human workers are prone to productivity degradation when ergonomics is not taken into account. This is true both when the humans work on their own and when they work with machines, such as collaborative robots.

One of the major issues regarding human ergonomics are improper working postures, which can produce excessive joint torques that are detrimental to the current task efficiency, as well

as to health and safety of the human in the long run (Keyserling and Chaffin, 1986; Kumar, 2001). Earlier methods of evaluating ergonomics of working postures, like *Rapid Upper Limb Assessment* (RULA) (McAtamney and Corlett, 1993) and *rapid entire body assessment* (REBA) (Hignett and McAtamney, 2000), used predefined heuristic tables that indicate a score of particular joint configuration. The combination of scores for all joints then gives the final score for the entire working configuration of an arm or a body. Recently, RULA and REBA have been applied to determine ergonomic postures in human-robot collaboration (Busch et al., 2017; Marin et al., 2018; Shafti et al., 2019). Other methods (Snook and Ciriello, 1991; Waters et al., 1993) used tables or equations to provide physical limits that should not be exceeded in terms of load during lifting tasks. Nevertheless, tables are more difficult to be personalized for a specific worker, and are more difficult to be generalized for different tasks and conditions. In addition, there are other important underlying indicators that affect the human ergonomics beyond kinematic posture and lifting load.

More recent methods included other indicators to optimize human working configuration, such as muscle comfort (Chen et al., 2018), physical fatigue (Maurice et al., 2016; Peternel et al., 2018b), energy consumption (Kim et al., 2010; Maurice et al., 2016), and arm manipulability (Jacquier-Bret et al., 2012; Gopinathan et al., 2017; Peternel et al., 2017; Petrič et al., 2019). In addition, they used personalized human body models (Maurice et al., 2014; Maurice et al., 2016; Kim et al., 2018a), which can be more easily integrated into collaborative robot controllers and generalized for many tasks. When a human is collaborating with a robot, we can use the robot to optimize the collaborative task execution based on the dynamical models of the human worker. Methods in Vahrenkamp et al. (2016) and Peternel et al. (2017) allowed the robot to plan the optimal handover of tools and objects between humans and robots by considering various factors, such as human dexterity and joint torques. Methods in Kim et al. (2018a) and Peternel et al. (2017) enabled the robot to detect the overloading joint torques in human body and then physically guide the human to change configuration online during the working process. Other methods in Peternel et al. (2018b, 2019) let the robot to estimate the human worker's muscle fatigue and then minimized it by reconfiguration of task execution. A similar method was also employed for ergonomic reconfiguration of human operator's arm in teleoperation (Peternel et al., 2020).

In Mansfeld et al. (2018), the authors proposed a concept called *Safety Map*, which used the information about robot inertia in different states of the workspace in combination with human injury data, to give workers a visual representation about the safety of interaction. Nevertheless, this map only examined safety in terms of possible collisions and gave no consideration to other major factors that affect the human ergonomics, such as joint torque, posture, and fatigue. Several methods used either one of these factors as ergonomics criterion (Kim et al., 2018a,b; Lorenzini et al., 2019; Peternel et al., 2019; Petrič et al., 2019). The method in Maurice et al. (2016) considered multiple criteria, but did not provide a combined overall ergonomics index. The methods in Peternel et al. (2017) and Chen et al. (2018) combined

two or more criteria to derive the optimal arm posture, but lacked a visual interface to convey the information about ergonomic suitability of the whole workspace.
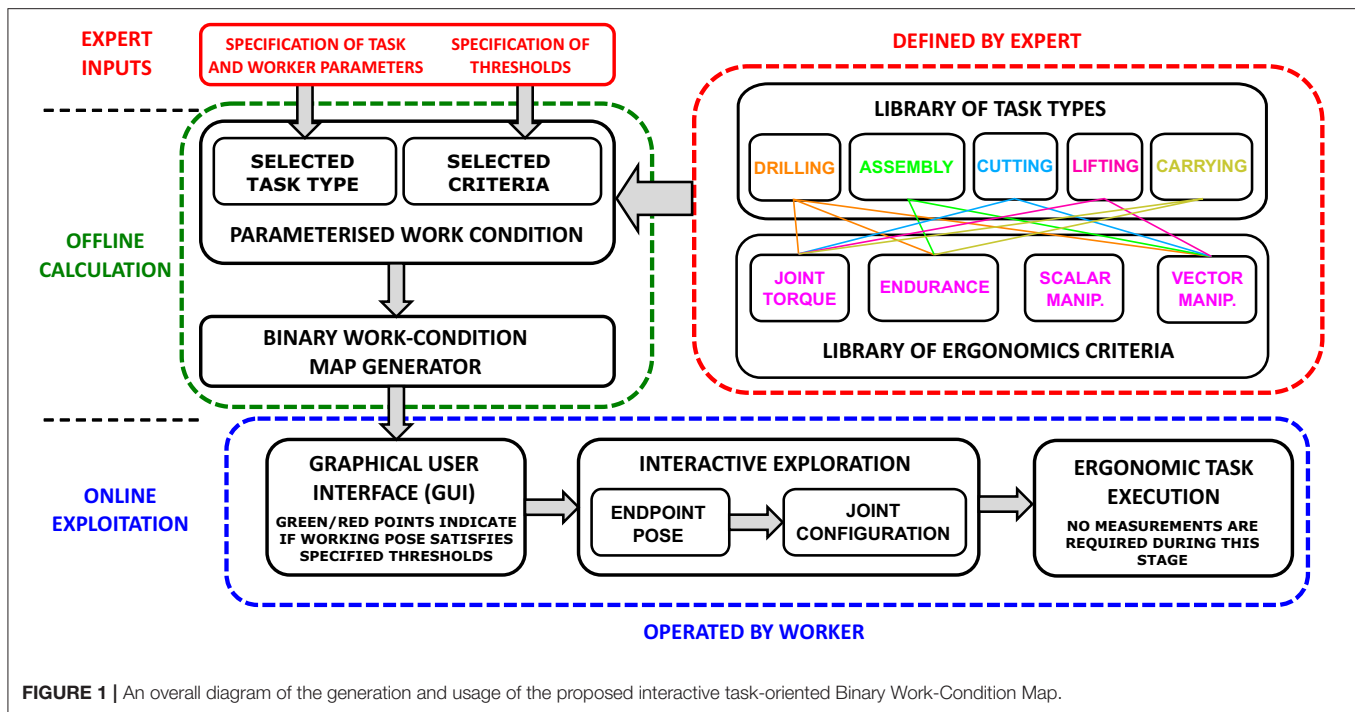
In Vahrenkamp et al. (2016), the authors proposed a concept called *Interaction Workspace*, which provided a visual color map of the workspace that indicated what arm postures are most suitable for task execution. Each posture had an index value that depended on a combination of several ergonomics criteria, such as human joint torque and dexterity. The index values were represented by color spectrum (i.e., one side of color spectrum for unfavorable values and the other side for favorable values). Nevertheless, the overall index for each posture was calculated by a weighted sum of all involved criteria (Vahrenkamp et al., 2016), therefore the contribution of each individual criterion may be unclear to users. Specifically, it may not be intuitive to a casual worker (and even experts) what a specific overall index value and its assigned color mean in terms of individual ergonomics factors. Moreover, due to the weighted sum, the overall index cannot guarantee that a given working posture does not exceed ergonomic thresholds of any individual criteria. These problems are also shared with RULA (McAtamney and Corlett, 1993) and REBA (Hignett and McAtamney, 2000), which provide a combined score from individual scores of different joints.

To resolve the above-mentioned issue, we propose a novel concept called *Binary Work-Condition Map*. Unlike methods that use weighted sum of various criteria (Vahrenkamp et al., 2016), the proposed method uses a threshold based approach for various criteria to obtain the overall ergonomics index at different positions of the workspace. This index is therefore binary and can potentially be more intuitive and easier to understand. For example if the index is one (logical true) in a given position, it means that all ergonomics criteria comply with the respective thresholds, which can be defined by the established safety and health standards and set by experts. If it is zero (logical false), then it is clear to a casual worker that a given working position does not satisfy all the standards and thresholds set by experts. In multi-color map (Vahrenkamp et al., 2016), this is not clear, because even in the safest "green" area, one of the thresholds might be exceeded, if the other factors are predominately satisfied due to the weighted-sum nature of derivation.

An additional contribution of the proposed work-condition map method is a novel display feature that can indicate ergonomic states of multiple arm postures sharing the same endpoint position for human arms, which possess such intrinsic kinematic redundancy. Such a feature is missing in the state-of-the-art work-condition map methods.

2. METHODS

An interactive Binary Work-Condition Map guides human workers to place their arms in appropriate postures for performing quasi-static manipulation tasks in an ergonomic and safe way. The method takes into account multiple task-related parameters and upper limb dynamic model (Saul et al., 2015) to create and if necessary update the binary map of suitable and non-suitable working postures. This map can be used to provide

FIGURE 1 | An overall diagram of the generation and usage of the proposed interactive task-oriented Binary Work-Condition Map.

human workers with posture guidance for accomplishing tasks independently. Alternatively, it can be shared with collaborative robots that are working together with human co-workers in order to optimize the collaboration. The workflow framework of the proposed interactive Binary Work-Condition Map is shown in **Figure 1**.

When a human worker is assigned to perform a specific task within the manufacturing process, the method categorizes the task by its type. The type of the task is associated with either a single criterion or multiple ergonomics criteria, which are used to evaluate various ergonomics and safety related work conditions during manipulation (e.g., joint torque, posture, fatigue, etc.). Different task types have different sets of associated criteria (see the top-right corner of **Figure 1**) and are represented in a library that is predefined based on the task knowledge provided by experts. For instance, in a task that requires a large force production for a short duration (e.g., lifting an object), the key criteria are joint torque and posture due to the large force, while fatigue is not dominant due to the short duration. In a task that requires a small force production for a long duration, fatigue becomes dominant and instantaneous joint torques play a lesser role.

The major types of criteria that we considered are joint torque, endurance, and manipulability. Instantaneous *joint torque* is important in terms of safety, as it can lead to various short-term and long-term injuries (Keyserling and Chaffin, 1986; Kumar, 2001; Kim et al., 2018a). On the other hand, integrated joint torque over time will lead to muscle fatigue (Peternel et al., 2018a,b, 2019), which can degrade the human worker's performance and endurance (De Luca, 1984; Enoka and Duchateau, 2008; Ma et al., 2009). The fatigue therefore

translates to *endurance time*, after which the worker cannot perform the required task production forces. The posture affects the manipulability of the human arm, which defines how well it can produces motion or forces at the endpoint (i.e., hand) in various directions of Cartesian space (Yoshikawa, 1985a,b; Petrič et al., 2019). The manipulability measure can be either scalar or vector. *Scalar manipulability* measure indicates how well the arm endpoint can produce both motion and force in all directions. For example, this measure is useful when the worker has to perform complex assembly tasks, where both motion and force are important in various directions to complete them. On the other hand, *vector manipulability* measure indicates how easy the arm endpoint can produce either motion or force in a specific direction of Cartesian space. For example, this measure is useful when the worker has to lift or carry a heavy object, where a good force production capability is necessary in the direction of the gravity.

Table 1 shows a library of considered tasks and criteria. For this study we considered five common manipulation tasks and four ergonomics criteria that are most relevant for manipulation tasks. Drilling task usually requires holding a heavy tool and producing relatively large forces for a prolonged time, therefore joint torque and endurance time are critical. Furthermore, the drilling force is in a specific direction, thus vector manipulability should be considered. Cutting and lifting require large effort as well, but only for a shorter time compared to drilling, therefore endurance time is not as important. On the other hand, carrying is typically a longer action than lifting and therefore requires endurance time consideration. Finally, a typical part assembly might not demand a lot of joint torque effort, however it may take a while thus endurance time is important. In addition, complex

	Joint torque	Endurance time	Scalar manipulability	Vector manipulability
Drilling	✔	✔	✗	✔
Assembly	✗	✔	✔	✗
Cutting	✔	✗	✗	✔
Lifting	✔	✗	✗	✔
Carrying	✔	✔	✗	✗

The marks ✔ and ✗ indicate whether a specific criterion is relevant for a specific task or not, respectively.

assembly tasks require producing forces and motion in various directions, therefore scalar manipulability is more important than vector manipulability. Note that this is a general framework and the considered tasks and criteria can be expanded. Certain criteria may become relevant or irrelevant depending on the specifics. For example, if an assembly task takes a lot of time, endurance time is relevant, while if it can be done quickly, it becomes irrelevant. When new tasks are added, a conceptual analysis is necessary to determine which ergonomics criteria are relevant.

When all the relevant criteria are determined according to the associated task type, the experts can set the thresholds for each ergonomics criteria. For example, if a certain amount torque is known to cause injuries and long-term health problems, the threshold is set conservatively below such limit. We should stress again that thresholding approach is the key difference compared to the existing methods that use weighted sum of criteria (Vahrenkamp et al., 2016). The proposed approach can guarantee that the preset thresholds for individual criteria will not be exceeded when the worker maintain the posture in the prescribed ergonomic area, while the continuous-condition maps obtained by a weighted sum cannot guarantee that.

The threshold can be set as a fixed limit below or above which the working arm posture is ergonomic as

$$c_i = \begin{cases} 1 & \text{if } v_i(\mathbf{q}, \mathbf{P}) < v_{th,i} \\ 0 & \text{if } v_i(\mathbf{q}, \mathbf{P}) \geq v_{th,i} \end{cases}, \tag{1}$$

where c_i is the binary index of i-th criterion, v_i is the i-th variable (e.g., joint torque, endurance time, etc.), and $v_{th,i}$ is the respective threshold. Variable v_i depends on arm posture that is defined by joint angles $\mathbf{q}$ and input parameters $\mathbf{P}$, which include task production force $\mathbf{f}_{ref}$ and other conditions. Note that inequality signs in (1) can be reversed, depending whether the more ergonomic state is below or above threshold. For example, in case of joint torque, the more ergonomic state is naturally below the threshold torque. In case of scalar manipulability, it is above the threshold since the larger manipulability is more ergonomic.

Alternatively, the threshold can be set as a range when the variable should be within some interval as

$$c_i = \begin{cases} 1 & \text{if } v_{min,i} < v_i(\mathbf{q}, \mathbf{P}) < v_{max,i} \\ 0 & \text{if } else \end{cases}, \tag{2}$$

where $v_{min,i}$ and $v_{max,i}$ are minimum and maximum threshold of the range, respectively. For example, (2) can be used instead of (1) when we want to make sure the joint torque does not exceed the safe limits (upper threshold v_{max}), but on the other hand, we do not want the worker to become too inactive (lower threshold v_{min}).

The parameters are passed on to the Binary Work-Condition Map generator that creates a workspace map for a given task by calculating a binary ergonomic state for each arm posture within the workspace as

$$e_k^{bin}(\mathbf{q}_k) = c_1 \wedge c_2 \wedge ...c_n, \tag{3}$$

where e_k^{bin} is the combined overall binary index for k-th human arm configuration $\mathbf{q}_k$, calculated by a logical AND operation $\wedge$ among the individual binary indices of various criteria c_i, $i = (1, 2, ...n)$. The considered criteria (joint torque, endurance time, and scalar and vector manipulability) and Binary Work-Condition Map generator are defined and described in the following subsections.

For comparison, the proposed threshold based approach is in contrast to the weighted-sum based approach in (Vahrenkamp et al., 2016), which produces a continuous ergonomic state for each arm posture as

$$e_k^{con}(\mathbf{q}_k) = c_1 w_1 + c_2 w_2 + ... c_n w_n, \tag{4}$$

where in this case criteria c_i have continuous values and w_i are their respective weights.

The advantage of the binary map is to be able to guarantee that the thresholds are met for all ergonomics criteria, however it has only binary states and better configurations among the good ones cannot be distinguished. On the other hand, the continuous map has more states and can therefore distinguish between different levels of good configurations, however it cannot guarantee that the thresholds for all criteria are met, even if the configuration is in the green section. That is because weighted-sum approach may produce a high score when majority of criteria are high, while one of them is very low and below a threshold.

In order to exploit the advantages of both the binary map and the continuous map, we also propose a novel hybrid map. In this approach, the binary map is used as a mask over the continuous map in order to filter out all configurations that do not meet the thresholds of all criteria. The remaining suitable sections of workspace are then colored by the continuous map in order to provide the user with a distinction between different levels of good configurations. The proposed hybrid map can be mathematically formalized as

$$e_k^{hyb}(\mathbf{q}_k) = e_k^{bin}(\mathbf{q}_k) \cdot e_k^{con}(\mathbf{q}_k), \tag{5}$$

In practice, the section of workspace that is red in the binary map remains red, while the green section can be recolored with a color scale to indicate multiple levels of goodness. Note that if applying very strict thresholds in the binary map, the border of the mask might be already in the green sections of the continuous map. To exploit the full color spectrum and better to distinguish different

levels of goodness for different points, the continuous sections of the hybrid map can be rescaled; for example, so that yellow color will start at the border instead of green color.

2.1. Joint Torque

The quasi-static relation between the human arm joint torques and endpoint force related to the task production is defined as

$$\tau = \mathbf{J}^T(\mathbf{q})\mathbf{f} + \mathbf{g}(\mathbf{q}), \tag{6}$$

where τ is the joint torque vector of the human arm, $\mathbf{f}$ is the endpoint force vector, $\mathbf{J}$ is the geometric Jacobian matrix of the human arm, $\mathbf{q}_h$ is the joint angle vector, and $\mathbf{g}_h$ is the gravity torque vector of the arm. As it can be seen from (6), the joint torque is affected by the force that is actively produced as a result of task performance, and the gravity of the human arm itself. The joint torque is calculated by using a human arm biomechanical model from Fang et al. (2018), which is based on the model developed in Holzbaur et al. (2005).

2.2. Endurance Time

Endurance time in which the human can perform the task with a specified force $\mathbf{f}$ is related to physical fatigue. We estimated the fatigue based on the model proposed in (Peternel et al., 2018b), which follows first-order system dynamics as other established models from the literature (Ma et al., 2009). Here we used joint torque as an effort estimation parameter as in (Maurice et al., 2016; Lamon et al., 2019). The fatigue model for each joint is defined as a first-order system of differential equations as

$$\frac{du_i(t)}{dt} = \begin{cases} \left(1 - u_i(t)\right)\frac{|\tau_i(\mathbf{q},t)|}{\lambda_i} & \text{if } |\tau_i(t)| \geq \tau_{th,i} \\ -u_i(t)\frac{R}{\lambda_i} & \text{if } |\tau_i(t)| < \tau_{th,i} \end{cases}, \tag{7}$$

where $u_i \in [0\ 1]$ is the i-th joint fatigue index, τ_i is calculated from (6) for a given time t and configuration $\mathbf{q}$, λ_i is a capacity parameter that determines the joint-specific fatigue characteristics. The parameters λ are dependent on individual human and joint. The higher the λ is, the more effort τ over time it takes for the fatigue to occur. The parameter R is a recovery rate, which determines the speed of fatigue reduction after the arm is relaxed. In our experiments we used a conservative value $R = 0.5$, as in (Peternel et al., 2018b) for all the joints. Other recovery rates can be found in literature (Ma et al., 2010). We used the threshold $\tau_{th,i}$ to determine when the arm joint is relaxed. When the joint torque is larger than this threshold, the model is in fatigue increasing mode, otherwise it is in recovery mode.

The values of fatigue capacity parameters λ of individual arm joints can be estimated by the method proposed in (Peternel et al., 2018b). In this procedure, the human produces several reference joint torques τ_{calib} for the amount of time T_{calib}, after which the human cannot endure it anymore or feels uncomfortable. In other words, one chooses τ_{calib} and measures respective T_{calib}. Capacity λ for each reference torque τ_{calib} is then derived by

$$\lambda = -\frac{|\tau_{calib}| \cdot T_{calib}}{\ln(1 - 0.993)}, \tag{8}$$

where the full capacity is assumed to be reached after five time constants, i.e., $u = 0.993$. The mean value of λ parameters, calculated by (8) for different reference forces, is then used as the final estimation of fatigue capacity for each joint separately.

The maximum endurance time T for an arbitrary joint torque τ is then obtained by,

$$T = -\frac{\lambda \cdot \ln(1 - 0.993)}{|\tau|}. \tag{9}$$

2.3. Scalar Manipulability

The scalar manipulability measures how well the arm endpoint can produce both motion and force in all direction of Cartesian space, and can be derived as Yoshikawa (1985b)

$$w = \sqrt{\det(\mathbf{J}(\mathbf{q})\mathbf{J}(\mathbf{q})^T)}, \tag{10}$$

where the higher value means more capacity to produce both motion and force at the endpoint. If the task requires complex manipulation that involves movements and force of the endpoint in various directions (e.g., complex assembly), it should be performed around the configuration where the manipulability w is the highest.

2.4. Vector Manipulability

The manipulability can also be examined on a vector level by using Eigen decomposition or singular value decomposing of arm Jacobian matrix (Yoshikawa, 1985b). Velocity manipulability is derived as

$$\mathbf{U}\Sigma\mathbf{V}^T = \mathbf{J}(\mathbf{q})\mathbf{J}(\mathbf{q})^T, \tag{11}$$

where Σ are singular values, while $\mathbf{U}$ and $\mathbf{V}$ are left and right singular vectors, respectively. Σ and $\mathbf{U}$ determine the size and shape of velocity manipulability ellipsoid, respectively. The size of a vector from the center of the ellipsoid to its surface in any direction tells how well the arm endpoint can move in that direction. Force manipulability is derived as

$$\mathbf{U}\Sigma\mathbf{V}^T = \left(\mathbf{J}(\mathbf{q})\mathbf{J}(\mathbf{q})^T\right)^{-1}, \tag{12}$$

where singular values and vectors have similar roles as in velocity manipulability. The force manipulability ellipsoid is able to tell how well the arm endpoint can produce or sustain forces in a certain direction.

The major axes of force and velocity manipulability are orthogonal; therefore if the arm in a given configuration can produce large velocities in a certain direction, then a large force cannot be produced in that direction, and vice-versa. For example, if the task requires to produce or sustain high forces in a certain direction (i.e., lifting a heavy object), the highest force manipulability vector should be aligned with that direction [i.e., (12) should be used]. If we need to move the manipulated object fast in a certain direction, the highest velocity manipulability vector can be aligned with that direction [i.e., (11) should be used]. In connection to the scalar manipulability from (10), high w tends to make velocity and force manipulability ellipsoids closer to a sphere.
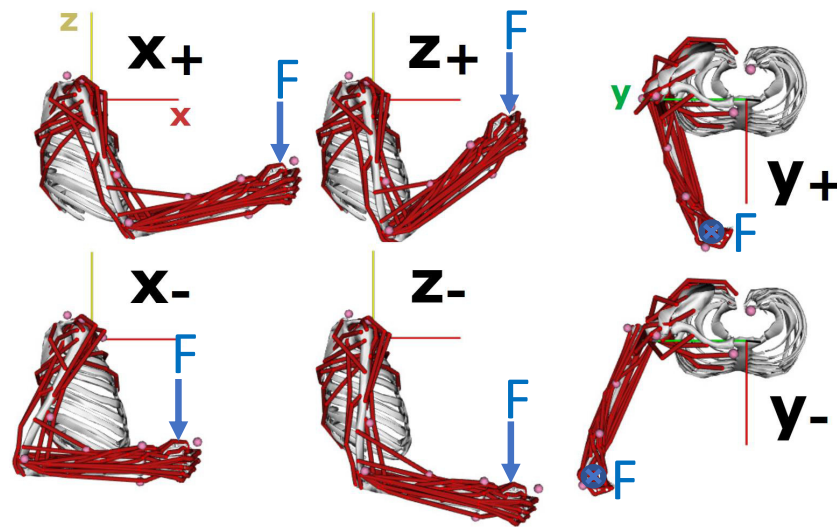
FIGURE 2 | Discretization of arm posture in terms of hand position in the proposed Binary Work-Condition Map by using human biomechanical model.

2.5. Binary Work-Condition Map Generator

To evaluate all possible arm postures, subject to selected workspace and joint limits, the whole configuration space of human arm is discretized in terms of the *Cartesian-Posture-Swivel-Angle* (CPSA) representation of human arm configuration (Fang et al., 2019). In this CPSA representation, a human arm configuration is expressed by a 3-degrees-of-freedom (DoF) position and 3-DoF orientation of human hand, plus 1-DoF swivel angle of the elbow, which is determined by the angle between a shoulder-elbow-wrist human arm plane and a vertical plane (Tolani et al., 2000).

Figure 2 provides an example of the discretization of arm posture in terms of hand (endpoint) position and **Figure 3** provides two examples of the discretization of the arm posture in terms of swivel angle. A hand position and swivel angle step sizes are predefined in advance. Every possible arm posture is tested under the desired external force required to produce the task (e.g., *F* shown in **Figure 2**). This is done automatically through an individual simulation using an OpenSim biomechanical model (Saul et al., 2015; Seth et al., 2018). The collected data for each posture is analyzed by (3) according to all the associated criteria and predefined thresholds. If all the criteria are satisfied and $e_k(\mathbf{q}_k) = 1$, the arm posture $\mathbf{q}_k$ will be labeled as a feasible configuration, otherwise it will be labeled as an infeasible posture.

After the offline calculation stage, the tested hand positions are visually presented to a worker through the developed graphical user interface (GUI) as a binary map of Cartesian points, where ergonomically feasible points are displayed by green color and infeasible by red color. Prior to the interactive exploration stage, the human worker attaches a set of pose-measurement markers on the anatomical landmarks of his/her arm. During the interactive exploration stage, the current arm posture is then captured by an optical motion capture system and reconstructed in a graphical user interface (GUI) in real time (Fang et al., 2018). This enables the worker to interactively explore the workspace

in real time through the GUI and generated map in the online exploitation stage.

Since there is a redundancy in the human arm, there are multiple possible configurations for a single hand position. The hand position is displayed as feasible (green color), if there is at least one feasible configuration within that hand position. The worker can then move the hand into that point and explore it by changing the configuration through swivel angle and redundant DoF in real time. If the configuration satisfies the criteria, the elbow point of the simulated human arm on the interactive map turns green, if not it turns red. Note that redundancy was not considered in (Vahrenkamp et al., 2016), therefore the proposed redundancy-display approach is novel in terms of interactive maps.

The examples of maps created for two arms of different dimensions are illustrated in **Figure 3**. Through interactive self-supervised exploration manner, the worker is able to establish an intuitive sense of how he/she should place the arm in appropriate configurations for performing the specified task. This self-supervised exploration can be further divided into practicing in the air and practicing with the real tool to help the user memorize the desirable arm configuration step by step. When a feasible arm posture is selected and memorized after the interactive exploration, the worker can execute the actual task without the assistance of the GUI and motion capture system.

3. EVALUATION AND RESULTS

The evaluation was separated into concept demonstration and experiments. The concept demonstration included demonstration of all aspects that does not include subjective factors of human worker (i.e., parts outside of blue area in **Figure 1**). These included technical calculation of work-condition maps taking a combination of different task parameters and criteria into account. The additional experiments

FIGURE 3 | A series of Binary Work-Condition Maps generated for five different tasks and two different human arm dimensions. Arm 1 had upper arm and forearm lengths 0.250 and 0.266 m, repetitively. Arm 2 had upper arm and forearm lengths 0.323 and 0.232 m, repetitively. X-axis point right of the body, y-axis points in front of the body and z-axis point vertically. The frame origin is at the right shoulder position. Green colored point indicates that if the endpoint is placed in it, at least one arm configuration exist where all criteria are within thresholds. Red colored point indicates that there are no ergonomic configurations in that position.

then evaluated aspects that involve subjective perception of human worker, i.e., usability factors of the developed binary and hybrid maps compared to the existing continuous map.
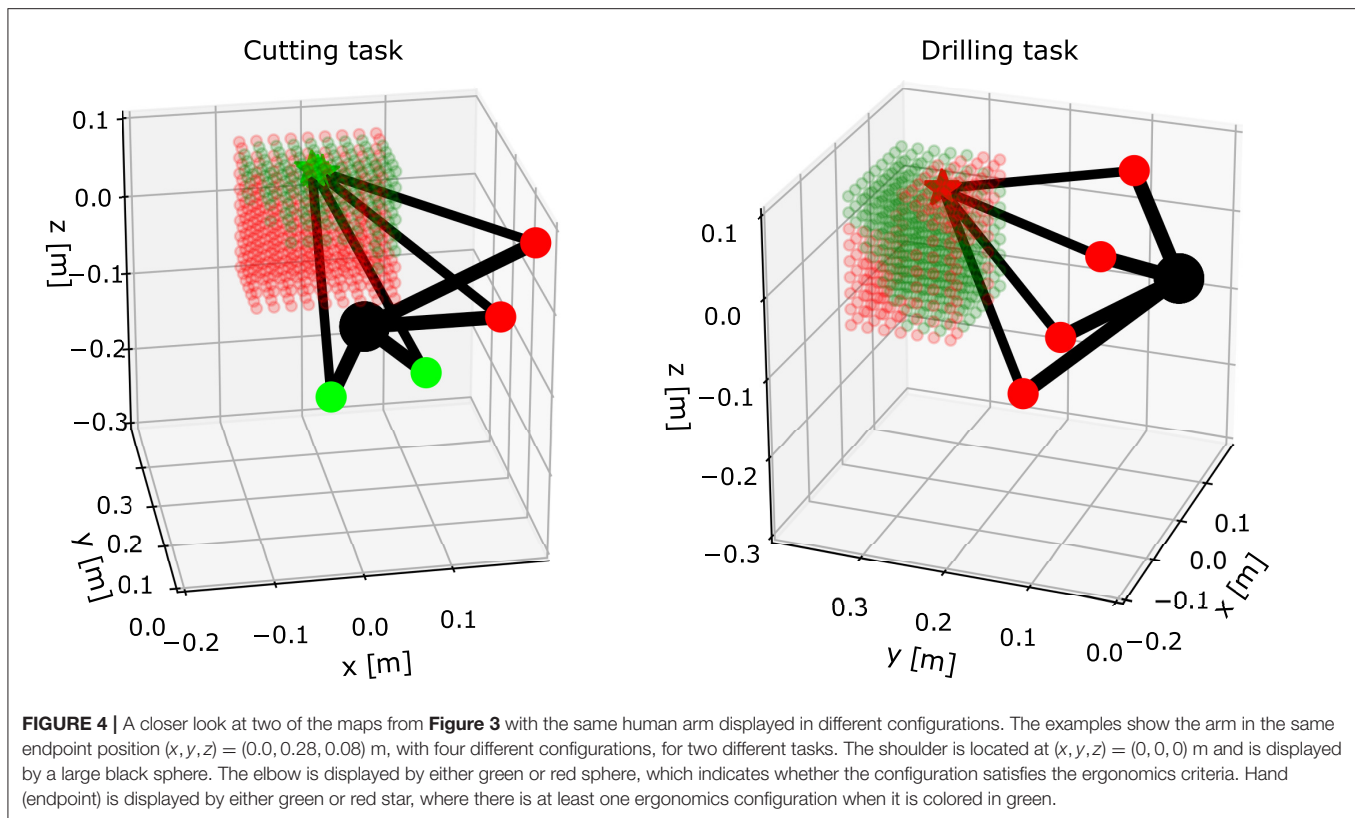
3.1. Concept Demonstration

To evaluate the offline part of the method, we calculated a Binary Work-Condition Map for each of the five considered tasks from the library in **Table 1**. For each map, the associated relevant ergonomics criteria were used to determine whether the available arm postures within the workspace are suitable or not. A 0.16 x 0.12 x 0.16 m cuboid in front of the body was selected as the workspace, with its center at $(x, y, z) = (0.0, 0.3, 0.0)$ m with respect to a base frame located at the right shoulder center. The positive x-axis of the base frame points rightwards from the shoulder, while the positive y-axis and z-axis points forwards and upwards, respectively. The condition was calculated for every point within the cuboid with resolution of 0.02 m. Note that the workspace and resolution can be adjusted depending on the scenario.

To demonstrate the effect of different arm dimensions on the calculation of the map, we generated five maps for two right arms of different dimensions. For the first human, the arm upper

arm length was 0.250 m and the forearm length was 0.266 m. For the second human, the arm upper arm length was 0.323 m and the forearm length was 0.232 m. We used one average and one extreme arm dimensions in order to highlight the conceptual differences.

Ten maps are produced based on the calculated results shown in **Figure 3**. By observing the maps on the figure, we can see that tasks have major influence on the map layout. For instance, the more demanding tasks in terms of physical effort, e.g., lifting and carrying (fourth and fifth rows), have very few arm configurations in green state that satisfy all selected ergonomics criteria. On the other hand, tasks that require less physical effort, like cutting (third row), have more arm configurations in green state. While complex assembly (second) is not a physically demanding task, it does have requirements from high manipulability; therefore areas, where the arm has to be extended, are in red state. Note that in order to highlight the differences between the tasks, we intentionally used relatively strict thresholds.

The influence of arm dimensions is also clearly visible by comparing the two columns. Different arm dimensions produced noticeably different values of scalar manipulability for assembly task (second row), and different values of velocity manipulability

FIGURE 4 | A closer look at two of the maps from **Figure 3** with the same human arm displayed in different configurations. The examples show the arm in the same endpoint position $(x, y, z) = (0.0, 0.28, 0.08)$ m, with four different configurations, for two different tasks. The shoulder is located at $(x, y, z) = (0, 0, 0)$ m and is displayed by a large black sphere. The elbow is displayed by either green or red sphere, which indicates whether the configuration satisfies the ergonomics criteria. Hand (endpoint) is displayed by either green or red star, where there is at least one ergonomics configuration when it is colored in green.

for the cutting task (third row). In addition, the maps for the drilling task were also considerably different because of a different combination of force manipulability and joint torque results due to different arm dimensions.

Figure 4 shows a closer look at two maps of the same human arm for two different tasks from **Figure 3**. Four arm configurations of evenly spaced swivel angles with the same endpoint position are displayed on each map (the resolution of swivel angle is 30 degrees). Since the work conditions are different because of the different tasks, the ergonomic states are different. In the cutting task (left plot) the endpoint position is in a green state, since there are two out of four configurations that satisfy all ergonomics criteria. Whether the configuration is ergonomic or not is indicated by green or red elbow, respectively. On the other hand, in the drilling task (right plot) the same endpoint position is in a red state, since there are no configurations that satisfy all ergonomics criteria. Therefore, all four configurations have red elbow.

Note that in this example we used four configurations for every endpoint position within the selected workspace. The amount of configurations per endpoint can be arbitrarily increased or decreased, depending on the use cases.
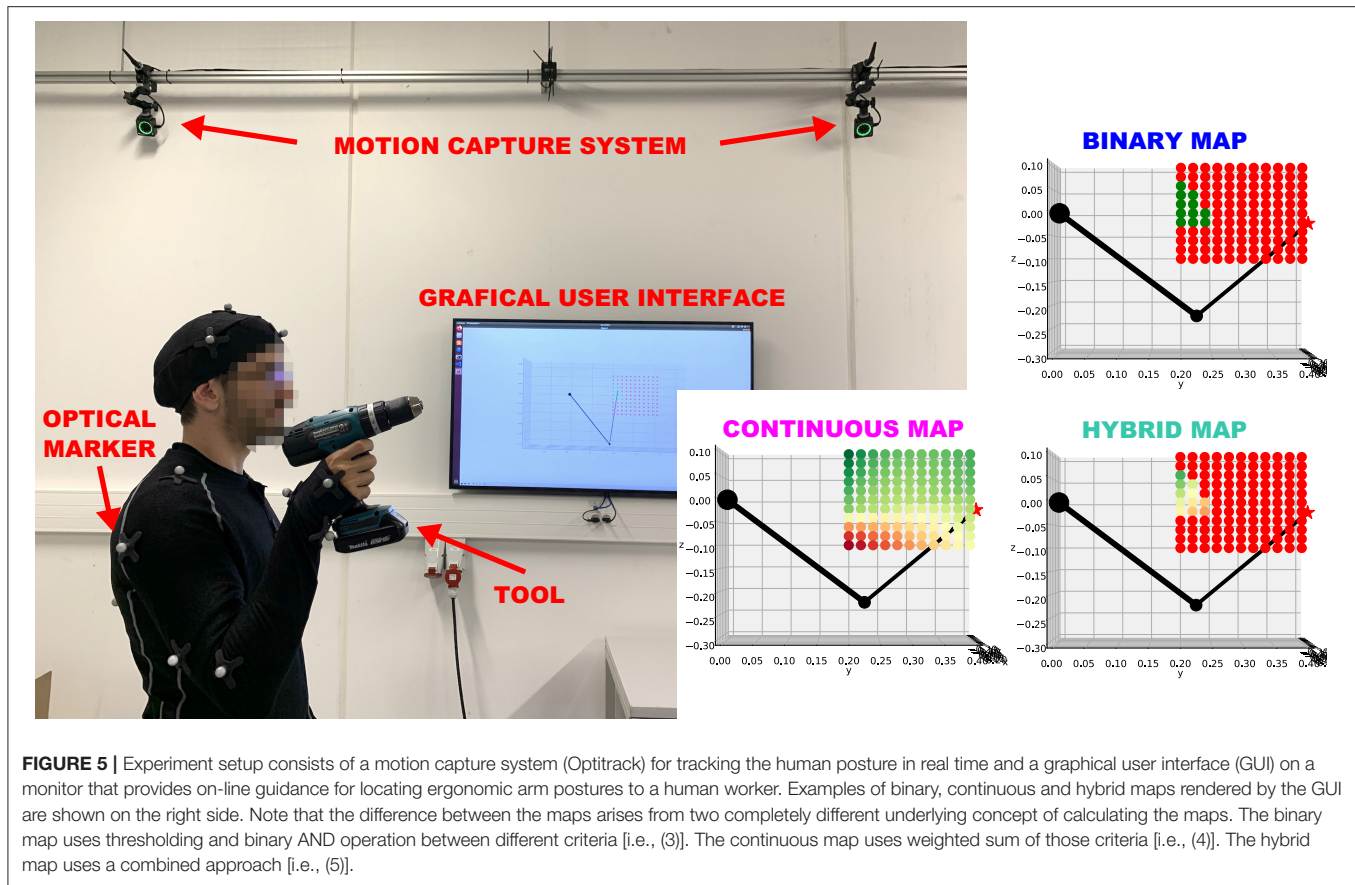
3.2. Experiments

The conceptual differences and advantages of the binary map compared to the continuous map were highlighted in section 2, and the main features of the proposed method were shown in section 3.1. Additionally, we performed experiments to compare

the different types of maps in terms of usability factors. The goal of the experiment was to evaluate subjective aspects of the proposed binary and hybrid-condition maps, compared to the continuous-condition map. We chose the continuous map as a benchmark in the comparison since it is a state-of-the-art method. Unlike the proposed method, the continuous map method (Vahrenkamp et al., 2016) did not consider redundant DoF of human arm and did not have any visualization solution for the redundant DoF. Therefore, in order to make a fair comparison, the experiments were performed using degenerate maps constrained on a 2D vertical plane, which is parallel to the human body sagittal plane and passes through the shoulder center.

We used 15 male participants in the experiment with age 27.60 ± 8.88 years, upper arm length 32.23 ± 1.85 cm and forearm length 27.93 ± 1.30 cm. The participants were briefed about the experiment procedure and the purpose of the experiment, and gave an informed consent to participation. We adapted the biomechanical model and parameters based on each individual participant during the calibration stage prior to the experiment.

The experiment setup (see **Figure 5**) included a motion capture system (Optitrack) that measured human arm configuration in real-time and a display (GUI) that showed the ergonomics maps with respect to the virtual copy of the human arm. The virtual copy of the arm moved in the same manner as the real arm according to the measured configuration.

Before the actual experiment, the participant conducted a familiarization experiment in order to get familiar with the

FIGURE 5 | Experiment setup consists of a motion capture system (Optitrack) for tracking the human posture in real time and a graphical user interface (GUI) on a monitor that provides on-line guidance for locating ergonomic arm postures to a human worker. Examples of binary, continuous and hybrid maps rendered by the GUI are shown on the right side. Note that the difference between the maps arises from two completely different underlying concept of calculating the maps. The binary map uses thresholding and binary AND operation between different criteria [i.e., (3)]. The continuous map uses weighted sum of those criteria [i.e., (4)]. The hybrid map uses a combined approach [i.e., (5)].

setup and the methods. The experimenter explained to the participant the implications of the different colors of the map in layman terms (e.g., the green color indicates good arm working posture and red indicates bad arm working posture). During the actual experiment the participant was instructed to explore workspace and to select an arbitrary suitable arm working posture. After their selection, the participant had to then produce the actual task, i.e., holding a heavy drilling tool at the selected position for 1 min. This procedure was sequentially done for all three maps. The order of maps as they were performed in the experiments was randomized between the participants.

After the experiments, we performed the subjective evaluation by a Likert-type of questionnaire, where the participant had to report the degree of agreement with the given statements:

- *S1*: The binary map is not ambiguous to indicate a good working posture.
- *S2*: The continuous map is not ambiguous to indicate a good working posture.
- *S3*: The hybrid map is not ambiguous to indicate a good working posture.
- *S4*: I feel it took effort to place my arm in a good configuration by binary map.
- *S5*: I feel it took effort to place my arm in a good configuration by continuous map.

- *S6*: I feel it took effort to place my arm in a good configuration by hybrid map.
- *S7*: I felt comfortable with the configuration selected by the binary map.
- *S8*: I felt comfortable with the configuration selected by the continuous map.
- *S9*: I felt comfortable with the configuration selected by the hybrid map.

There were five possible levels of agreement (score is in the brackets): strongly agree (2), agree (1), neutral (0), disagree (−1), strongly disagree (−2). *S1–S3* evaluate the initial phase of the method, where the user has to visually search for and select a suitable configuration on a given map. *S4–S6* evaluate the middle phase of the method, where the user has to explore and navigate to the selected configuration. *S7–S9* evaluate the final phase of the method, where the user has to perform the task in the selected configuration. Additionally, we asked the participants to rank the methods according to their overall preference, where 3 points were given to the best and 1 point the worse method in terms of preference.

To check for significance of the differences between subjective scores for the three methods, we performed a statistical analysis using paired sample *t*-tests. The statistical significance was set to 0.05 and statistical power to 0.8. Power analysis indicated that sample number of 14 was sufficient under the given parameters.
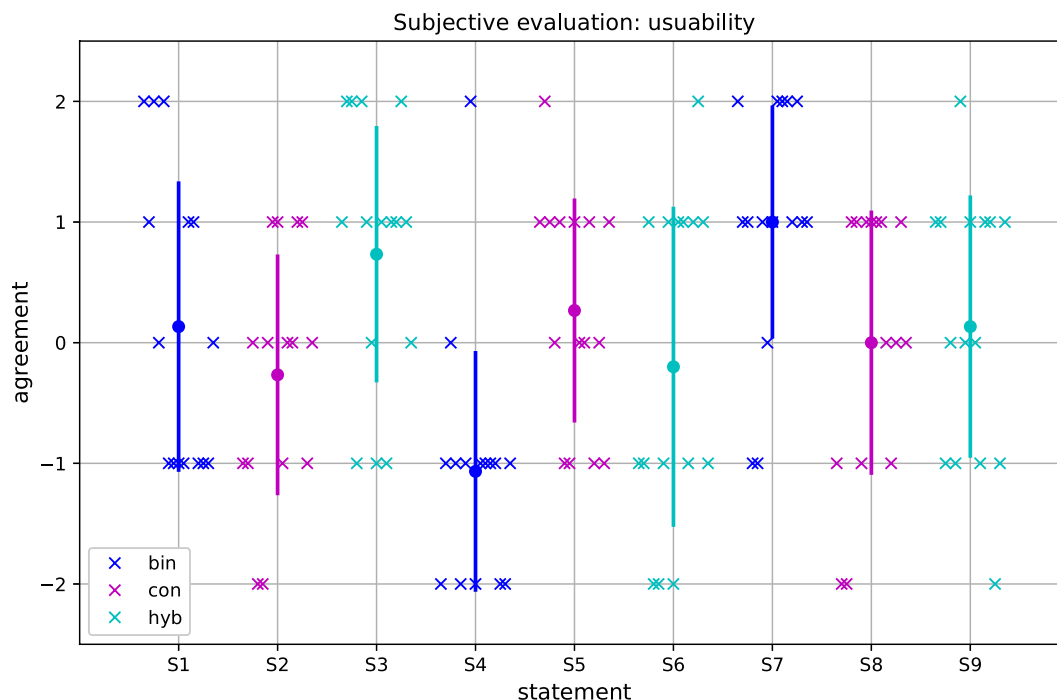
FIGURE 6 | Results of subjective evaluation for usability. The statements are listed on x-axis, while the level of agreement in terms of score is shown on y-axis. Blue color is associated with the statements related to the binary map, magenta color is associated with the statements related to the continuous map and cyan color is associated with the statements related to the hybrid map. The dot represents mean score and the vertical line represents standard deviation. The individual data points are marked by crosses. In essence, positive scores for *S1–S3* indicate that the method was not ambiguous. On the other hand, negative scores for *S4–S6* indicate that the method took less effort to use. Finally, positive scores for *S7–S9* indicate that the task execution in the selected configuration was comfortable.

TABLE 2 | Results of subjective analysis of usability.

Aspect	Statement	Agreement
Not ambiguous to	S1 (binary)	0.13±1.20
indicate a good	S2 (contin)	-0.27±1.00
working posture.	S3 (hybrid)	0.73±1.06*
It took effort to	S4 (binary)	-1.07±1.00*
place my arm in a	S5 (contin)	0.27±0.93
good configuration.	S6 (hybrid)	–0.20±1.33
	S7 (binary)	1.00±0.97*
Comfortable with the	S8 (contin)	0.00±1.10
selected configuration.	S9 (hybrid)	0.13±1.09

The values represent the mean degree of agreement to the statements S1–S9 and respective standard deviation. Positive value indicates agreement, while negative indicates disagreement with the statement for a given map. Symbol * indicates whether there is a significant difference with respect to the continuous map (benchmark).

The datasets were checked for normality by performing the Shapiro-Wilk test. If the dataset did not pass this test, non-normal distributed data was corrected by a rank-transformation before the main test.
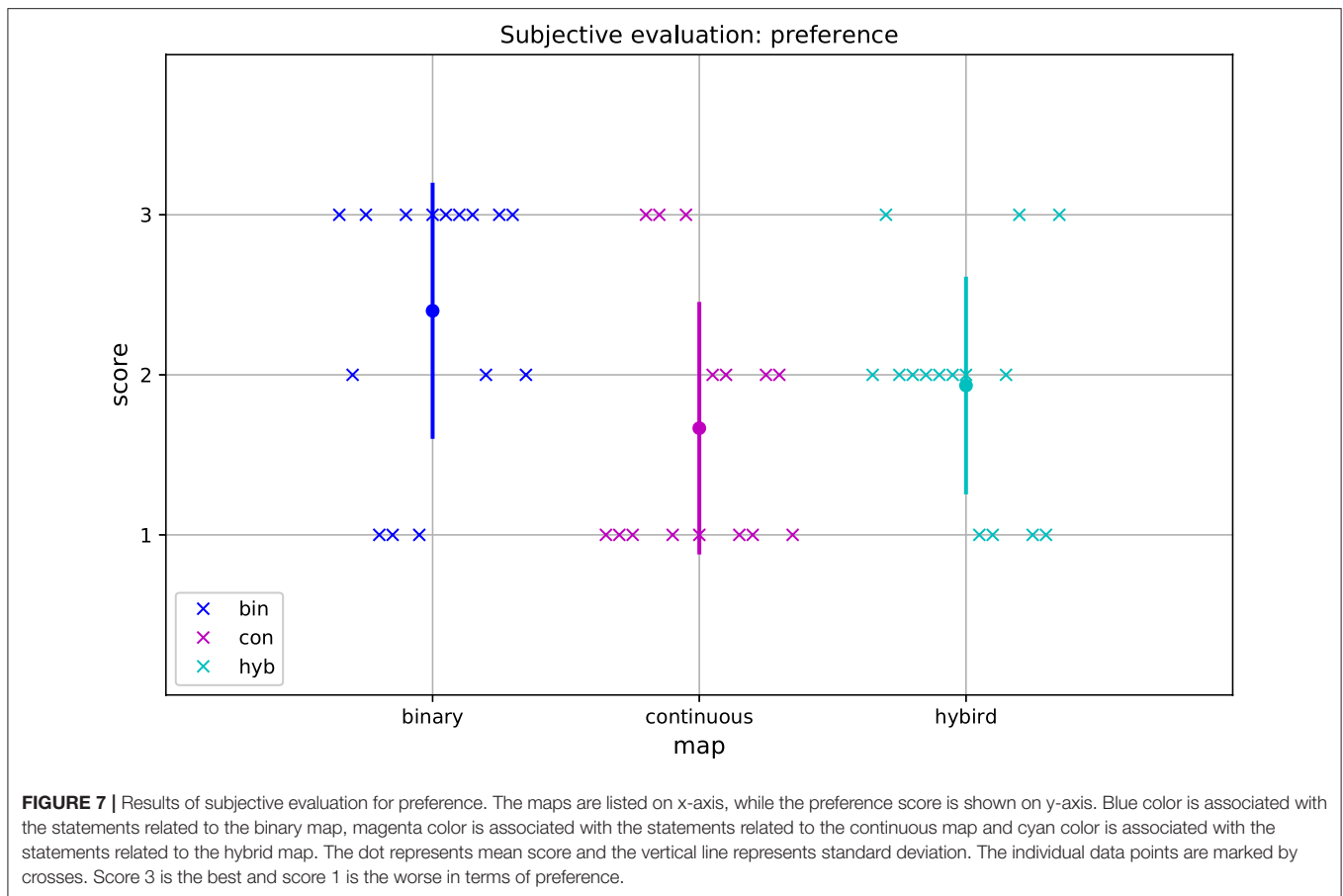
The results of the subjective evaluation for usability are shown in **Figure 6**. Additionally, the average degree of agreement to

the statements *S1–S9* is shown is **Table 2**. On average, the participants rated the binary map less ambiguous compared to the continuous map. However, the difference was not statistically significant ($p = 0.442$). The hybrid map was also rated less ambiguous compared to the continuous map. The difference was statistically significant ($p = 0.030$).

On average, the participants felt that by using the continuous map it took much more effort to explore and navigate to the selected configuration compared to the binary map. The difference was statistically significant ($p = 0.006$). The same was true for the hybrid map when compared to the continuous map. However, the difference was not statistically significant ($p = 0.389$).

On average, the participants felt that performing the task in the selected configuration, by using the binary, was more comfortable compared to the one selected by the continuous map. The difference was statistically significant ($p = 0.038$). The same was true for the hybrid map when compared to the continuous map. However, the difference was statistically not significant ($p = 0.784$).

The results of the subjective evaluation for overall preference of maps are shown in **Figure 7**. The participants generally preferred the binary map to either the hybrid map or the continuous map, and most of them gave the binary map the highest score. However, the preference difference between

FIGURE 7 | Results of subjective evaluation for preference. The maps are listed on x-axis, while the preference score is shown on y-axis. Blue color is associated with the statements related to the binary map, magenta color is associated with the statements related to the continuous map and cyan color is associated with the statements related to the hybrid map. The dot represents mean score and the vertical line represents standard deviation. The individual data points are marked by crosses. Score 3 is the best and score 1 is the worse in terms of preference.

the binary map and the continuous map was statistically not significant ($p = 0.077$). Neither was significant the difference between the hybrid map and the continuous ($p = 0.433$).

4. DISCUSSION

The main strength and advantage of the binary map compared to the continuous map is that it can guarantee the thresholds for all criteria are satisfied. This is not the case with the continuous map, since it uses weighted sum to derive the overall index for a given position. For example, the continuous map does not guarantee that all criteria are within the ergonomic thresholds, even in the high-value points (green color). This is a conceptual advantage which was highlighted in the section 2. The hybrid map uses the binary map concept to rule out the configurations that do not satisfy all the thresholds; therefore it exploits the main conceptual advantage of the binary map. On the other hand, it uses continuous map concept to indicate the different levels of ergonomy among the suitable ones in order to increase the resolution for the user.

The results of experiments and subjective evaluation showed that the participants found the binary map and the hybrid map less ambiguous, compared to the continuous map. This can be mostly likely attributed to the binary nature of the map, since the map gives two distinct states and therefore it is clear to

the user whether the posture is either ergonomic or not (i.e., whether the thresholds set by the expert are satisfied or not). On the other hand, the continuous map has multiple states and gives a range of ergonomic values, which can be ambiguous. Medium value (yellow color) in the continuous map can be achieved by different combinations of criteria conditions, for example: non-ergonomic torque and ergonomic manipulability, or ergonomic torque and non-ergonomic manipulability, or borderline ergonomic for both torque and manipulability. Just by looking at the map, it is impossible even for an expert to know for sure which combination produced the given color, let alone a casual worker. However, the hybrid map was rated even less ambiguous which might be attributed to the exploitation of advantages of both the binary and the continuous map; a clear division between unsuitable and suitable configurations areas, but a continuous pattern within the suitable ones that permits more resolution in the selection of the best among the good ones. Therefore, this hints that the hybrid map and the binary map might be more suitable for casual workers that are not experts in ergonomics.

The results of subjective evaluation also showed that the participants felt it took less effort to use the binary map to explore and navigate to the selected configuration, compared to the continuous map. This could potentially be attributed to the binary map concept that rules out a considerable number

of configurations for not satisfying all the thresholds. The continuous map concept has continuous states across all the workspace and therefore more options to navigate through. In addition, more continuous states can take more attention from the user in order to distinguish between the different tones of color while exploring through the map. As the results showed, the difference in perceived effort between hybrid map and the continuous map is not as large, which can be attributed to hybrid map taking aspects from both the binary map and the continuous map. Based on this, we recommend using the binary map when new working configurations are changing quickly in order to minimize the perceived workload on the user. If that is not the case, we recommend using the hybrid map in order to exploit also the advantage of the continuous map.

The perceived better comfort in the configurations selected by the binary map can be attributed to the binary map guaranteeing that all ergonomic thresholds are being met through the underlying thresholding approach. On the other hand, the continuous map uses weighted-sum approach and does not guarantee that all ergonomic thresholds are satisfied, even for highly rated configurations. We recommend using the binary map when satisfying thresholds for all criteria is of primary importance.

Surprisingly, the participants did not perceive the same comfort difference for the hybrid map. Different participants might subjectively weigh different relevant criteria in different ways (not equally), however equal weights among different criteria were assumed in the experiments for the hybrid and continuous maps. By using the binary map, the participants could choose the configuration, which they felt it is the most ergonomic, among several options in the green area. This implies that they might have potentially used their "embedded" non-equal weights to explore and search for their own customized best configuration in the binary map. On the other hand, when using the hybrid map, the equal weights among different criteria were hard-coded and inherited from the continuous map.

Finally, the participants on average preferred the binary map in overall sense. This might be attributed to the clear and easy-to-read distinction between the suitable and unsuitable working configurations by the binary states. Nevertheless, the differences for the preference were not statistically significant; therefore we recommend that subjective preference should be examined individually for a specific user.

5. CONCLUSION

In conclusion, we recommend that the selection of map should be primarily based on the different advantages of the maps with respect to the specific requirements of a given application. If maintaining the thresholds strictly is important, we recommend using the binary map. Subjective aspects can be considered as secondary reason for selection. For example, if easy-to-use aspect is important, we recommend the binary map. If higher resolution of states is require, the continuous map provides such intrinsic advantage. Nevertheless, in such case we recommend using the hybrid map instead of the pure continuous map, since it combines the advantages of the binary and continuous maps, at a slight expense of complexity compared to the binary map.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/supplementary materials, further inquiries can be directed to the corresponding author/s.

ETHICS STATEMENT

Ethical review and approval was not required for the study on human participants in accordance with the local legislation and institutional requirements. The patients/participants provided their written informed consent to participate in this study.

AUTHOR CONTRIBUTIONS

LP and CF developed the concept and methods. LP, DS, and CF contributed to programming, experiments and data analysis. LP wrote the first draft of the paper. LP and CF revised the paper. All authors read and approved the submitted version.

FUNDING

REFERENCES

Busch, B., Maeda, G., Mollard, Y., Demangeat, M., and Lopes, M. (2017). "Postural optimization for an ergonomic human-robot interaction," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Vancouver, BC: IEEE), 2778–2785.

Chen, L., Figueredo, L. F., and Dogar, M. R. (2018). "Planning for muscular and peripersonal-space comfort during human-robot forceful collaboration," in *2018 IEEE-RAS 18th International Conference on Humanoid Robots (Humanoids)* (Beijing: IEEE), 1–8.

De Luca, C. J. (1984). Myoelectrical manifestations of localized muscular fatigue in humans. *Crit. Rev. Biomed. Eng.* 11, 251–279.

Enoka, R. M., and Duchateau, J. (2008). Muscle fatigue: what, why and how it influences muscle function. *J. Physiol.* 586, 11–23. doi: 10.1113/jphysiol.2007.139477

Fang, C., Ajoudani, A., Bicchi, A., and Tsagarakis, N. G. (2018). "A real-time identification and tracking method for the musculoskeletal model of human arm," in *2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC)* (Miyazaki: IEEE), 3472–3479.

Fang, C., Ding, X., Zhou, C., and Tsagarakis, N. (2019). A2ml: a general human-inspired motion language for anthropomorphic arms based on movement primitives. *Robot. Auton. Syst.* 111, 145–161. doi: 10.1016/j.robot.2018.10.006

Gopinathan, S., Ötting, S. K., and Steil, J. J. (2017). A user study on personalized stiffness control and task specificity in physical human–robot interaction. *Front. Robot. AI* 4:58. doi: 10.3389/frobt.2017.00058

Hignett, S., and McAtamney, L. (2000). Rapid entire body assessment (reba). *Appl. Ergon.* 31, 201–205. doi: 10.1016/S0003-6870(99)00039-3

Holzbaur, K. R., Murray, W. M., and Delp, S. L. (2005). A model of the upper extremity for simulating musculoskeletal surgery and analyzing neuromuscular control. *Ann. Biomed. Eng.* 33, 829–840. doi: 10.1007/s10439-005-3320-7

Jacquier-Bret, J., Gorce, P., and Rezzoug, N. (2012). The manipulability: a new index for quantifying movement capacities of upper extremity. *Ergonomics* 55, 69–77. doi: 10.1080/00140139.2011.633176

Keyserling, W. M., and Chaffin, D. B. (1986). Occupational ergonomics-methods to evaluate physical stress on the job. *Annu. Rev. Public Health* 7, 77–104. doi: 10.1146/annurev.pu.07.050186.000453

Kim, W., Lee, J., Peternel, L., Tsagarakis, N., and Ajoudani, A. (2018a). Anticipatory robot assistance for the prevention of human static joint overloading in human–robot collaboration. *IEEE Robot. Automat. Lett.* 3, 68–75. doi: 10.1109/LRA.2017.2729666

Kim, W., Lee, S., Kang, M., Han, J., and Han, C. (2010). "Energy-efficient gait pattern generation of the powered robotic exoskeleton using dme," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems* (Taipei: IEEE), 2475–2480.

Kim, W., Lorenzini, M., Kapıcıoğlu, K., and Ajoudani, A. (2018b). Ergotac: a tactile feedback interface for improving human ergonomics in workplaces. *IEEE Robot. Autom. Lett.* 3, 4179–4186. doi: 10.1109/LRA.2018.2864356

Kumar, S. (2001). Theories of musculoskeletal injury causation. *Ergonomics* 44, 17–47. doi: 10.1080/00140130120716

Lamon, E., De Franco, A., Peternel, L., and Ajoudani, A. (2019). A capability-aware role allocation approach to industrial assembly tasks. *IEEE Robot. Autom. Lett.* 4, 3378–3385. doi: 10.1109/LRA.2019.2926963

Lorenzini, M., Kim, W., De Momi, E., and Ajoudani, A. (2019). "A new overloading fatigue model for ergonomic risk assessment with application to human-robot collaboration," in *2019 International Conference on Robotics and Automation (ICRA)* (Montreal, QC: IEEE), 1962–1968.

Ma, L., Chablat, D., Bennis, F., and Zhang, W. (2009). A new simple dynamic muscle fatigue model and its validation. *Int. J. Indus. Ergon.* 39, 211–220. doi: 10.1016/j.ergon.2008.04.004

Ma, L., Chablat, D., Bennis, F., Zhang, W., and Guillaume, F. (2010). A new muscle fatigue and recovery model and its ergonomics application in human simulation. *Virt. Phys. Prototyping* 5, 123–137. doi: 10.1080/17452759.2010.504056

Mansfeld, N., Hamad, M., Becker, M., Marin, A. G., and Haddadin, S. (2018). Safety map: a unified representation for biomechanics impact data and robot instantaneous dynamic properties. *IEEE Robot. Autom. Lett.* 3, 1880–1887. doi: 10.1109/LRA.2018.2801477

Marin, A. G., Shourijeh, M. S., Galibarov, P. E., Damsgaard, M., Fritzsche, L., and Stulp, F. (2018). "Optimizing contextual ergonomics models in human-robot interaction," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Madrid: IEEE), 1–9.

Maurice, P., Padois, V., Measson, Y., and Bidaud, P. (2016). Experimental assessment of the quality of ergonomic indicators for dynamic systems computed using a digital human model. *Int. J. Hum. Fact. Model. Simul.* 5, 190–209. doi: 10.1504/IJHFMS.2016.10000531

Maurice, P., Schlehuber, P., Padois, V., Measson, Y., and Bidaud, P. (2014). "Automatic selection of ergonomie indicators for the design of collaborative robots: a virtual-human in the loop approach," in *2014 IEEE-RAS International Conference on Humanoid Robots* (Madrid), 801–808.

McAtamney, L., and Corlett, E. N. (1993). Rula: a survey method for the investigation of work-related upper limb disorders. *Appl. Ergon.* 24, 91–99. doi: 10.1016/0003-6870(93)90080-S

Peternel, L., Fang, C., Laghi, M., Bicchi, A., Tsagarakis, N., and Ajoudani, A. (2020). "Human arm posture optimisation in bilateral teleoperation through interface reconfiguration," in *2020 8th IEEE RAS/EMBS International Conference for Biomedical Robotics and Biomechatronics (BioRob)* (New York, NY: IEEE), 1102–1108.

Peternel, L., Fang, C., Tsagarakis, N., and Ajoudani, A. (2018a). "Online human muscle force estimation for fatigue management in human-robot co-manipulation," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Madrid: IEEE), 1340–1346.

Peternel, L., Fang, C., Tsagarakis, N., and Ajoudani, A. (2019). A selective muscle fatigue management approach to ergonomic human-robot co-manipulation. *Robot. Comput. Integr. Manuf.* 58, 69–79. doi: 10.1016/j.rcim.2019.01.013

Peternel, L., Kim, W., Babič, J., and Ajoudani, A. (2017). "Towards ergonomic control of human-robot co-manipulation and handover," in *2017 IEEE-RAS 17th International Conference on Humanoid Robotics (Humanoids)* (Birmingham), 55–60.

Peternel, L., Tsagarakis, N., Caldwell, D., and Ajoudani, A. (2018b). Robot adaptation to human physical fatigue in human–robot co-manipulation. *Auton. Robots* 42, 1011–1021. doi: 10.1007/s10514-017-9678-1

Petrič, T., Peternel, L., Morimoto, J., and Babič, J. (2019). Assistive arm-exoskeleton control based on human muscular manipulability. *Front. Neurorobot.* 13:30. doi: 10.3389/fnbot.2019.00030

Saul, K. R., Hu, X., Goehler, C. M., Vidt, M. E., Daly, M., Velisar, A., et al. (2015). Benchmarking of dynamic simulation predictions in two software platforms using an upper limb musculoskeletal model. *Comput. Methods Biomech. Biomed. Eng.* 18, 1445–1458. doi: 10.1080/10255842.2014.916698

Seth, A., Hicks, J. L., Uchida, T. K., Habib, A., Dembia, C. L., Dunne, J. J., et al. (2018). Opensim: Simulating musculoskeletal dynamics and neuromuscular control to study human and animal movement. *PLoS Comput. Biol.* 14:e1006223. doi: 10.1371/journal.pcbi.1006223

Shafti, A., Ataka, A., Lazpita, B. U., Shiva, A., Wurdemann, H. A., and Althoefer, K. (2019). "Real-time robot-assisted ergonomics," in *2019 International Conference on Robotics and Automation (ICRA)* (Montreal, QC: IEEE), 1975–1981.

Snook, S. H., and Ciriello, V. M. (1991). The design of manual handling tasks: revised tables of maximum acceptable weights and forces. *Ergonomics* 34, 1197–1213. doi: 10.1080/00140139108964855

Tolani, D., Goswami, A., and Badler, N. I. (2000). Real-time inverse kinematics techniques for anthropomorphic limbs. *Graph. Models* 62, 353–388. doi: 10.1006/gmod.2000.0528

Vahrenkamp, N., Arnst, H., Wachter, M., Schiebener, D., Sotiropoulos, P., Kowalik, M., et al. (2016). "Workspace analysis for planning human-robot interaction tasks," in *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)* (Cancun), 1298–1303.

Waters, T. R., Putz-Anderson, V., Garg, A., and Fine, L. J. (1993). Revised niosh equation for the design and evaluation of manual lifting tasks. *Ergonomics* 36, 749–776. doi: 10.1080/00140139308967940

Yoshikawa, T. (1985a). "Dynamic manipulability of robot manipulators," in *Robotics and Automation. Proceedings. 1985 IEEE International Conference on*, Vol. 2 (St. Louis, MO: IEEE), 1033–1038.

Yoshikawa, T. (1985b). Manipulability of robotic mechanisms. *Int. J. Robot. Res.* 4, 3–9. doi: 10.1177/027836498500400201

Advantages of publishing in Frontiers

OPEN ACCESS
Articles are free to read for greatest visibility and readership

FAST PUBLICATION
Around 90 days from submission to decision

HIGH QUALITY PEER-REVIEW
Rigorous, collaborative, and constructive peer-review

TRANSPARENT PEER-REVIEW
Editors and reviewers acknowledged by name on published articles

Frontiers
Avenue du Tribunal-Fédéral 34
1005 Lausanne | Switzerland

Visit us: www.frontiersin.org
Contact us: frontiersin.org/about/contact

REPRODUCIBILITY OF RESEARCH
Support open data and methods to enhance research reproducibility

DIGITAL PUBLISHING
Articles designed for optimal readership across devices

FOLLOW US
@frontiersin

IMPACT METRICS
Advanced article metrics track visibility across digital media

EXTENSIVE PROMOTION
Marketing and promotion of impactful research

LOOP RESEARCH NETWORK
Our network increases your article's readership