frontiers in Neurorobotics

ORIGINAL RESEARCH
published: 14 February 2020
doi: 10.3389/fnbot.2019.00117

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

OPEN ACCESS

Edited by:
Bin Fang,
Tsinghua University, China

Reviewed by:
Zhen Deng,
Universität Hamburg, Germany
Rui Chen,
Chongqing University, China
Haiming Huang,
Shenzhen University, China

***Correspondence:**
Wusheng Chou
wschou@buaa.edu.cn

Received: *26 August 2019*
Accepted: *24 December 2019*
Published: *14 February 2020*

Citation:
Jiao R, Wang Z, Chu R, Dong M, Rong Y and Chou W (2020) An Intuitive End-to-End Human-UAV Interaction System for Field Exploration. Front. Neurorobot. 13:117. doi: 10.3389/fnbot.2019.00117

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:

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

The sign function can be described as:

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

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:

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

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

$$\boldsymbol{C_A^B} = (q_0^2 - \boldsymbol{q_v}^T\boldsymbol{q_v})\boldsymbol{I_3} + 2\boldsymbol{q_v}\boldsymbol{q_v}^T + 2q_0S(\boldsymbol{q_v}) \tag{4}$$

The time derivative of Equation (4) is:

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

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:

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

$$\boldsymbol{q_e} = \boldsymbol{q_d^*} \otimes \boldsymbol{q} \tag{7}$$

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.

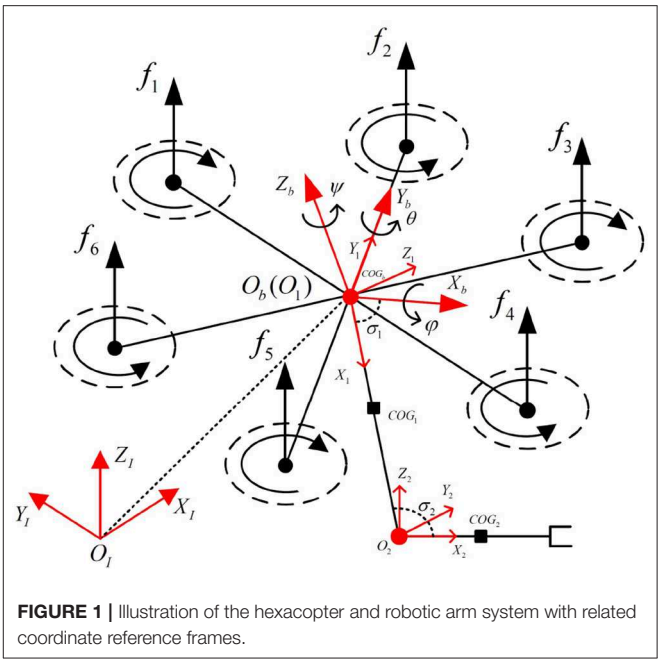

FIGURE 1 | Illustration of the hexacopter and robotic arm system with related coordinate reference frames.

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:

$$\boldsymbol{\omega} = \boldsymbol{R}_r\dot{\boldsymbol{\Psi}} \tag{8}$$

where

$$\boldsymbol{R}_r = \begin{bmatrix} 1 & 0 & -s\theta \\ 0 & c\varphi & s\varphi c\theta \\ 0 & -s\varphi & c\varphi c\theta \end{bmatrix} \tag{9}$$

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:

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

where

$$\begin{cases} c_1 = y_G(\dot{v}_z - v_x\omega_y + v_y\omega_x) - z_G(\dot{v}_y - v_z\omega_x + v_x\omega_z) \\ c_2 = -x_G(\dot{v}_z - v_x\omega_y + v_y\omega_x) + z_G(\dot{v}_x - v_y\omega_z + v_z\omega_y) \\ c_3 = -y_G(\dot{v}_x - v_y\omega_z + v_z\omega_y) + x_G(\dot{v}_y - v_z\omega_x + v_x\omega_z) \end{cases} \tag{11}$$

We describe Equation (11) in a collective form:

$$\boldsymbol{J}\dot{\boldsymbol{\omega}} = \boldsymbol{u} - \boldsymbol{S}(\boldsymbol{\omega})\boldsymbol{J}\boldsymbol{\omega} - m\boldsymbol{c} + \boldsymbol{d} \tag{12}$$

where vector $\boldsymbol{J} = diag(J_x, J_y, J_z)$ indicates that the inertia matrix is diagonal, and $\boldsymbol{d} = [d_x, d_y, d_z]^T$ denotes the lumped disturbances caused by wind gusts, model uncertainties, etc. The COG of the whole UAV system is described by $\boldsymbol{C_G} = [x_G, y_G, z_G]^T$. m is the total mass of the UAV. Additionally, we define vector $\boldsymbol{c} = [c_1, c_2, c_3]^T$ and vector $\boldsymbol{u} = [u_1, u_2, u_3]^T$, 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:

$$\boldsymbol{u} = \boldsymbol{\Xi}\boldsymbol{f_v} \tag{13}$$

where $\boldsymbol{f_v} = [\omega_1^2, \omega_2^2, \omega_3^2, \omega_4^2, \omega_5^2, \omega_6^2]^T$ 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**, $\boldsymbol{\Xi}$ can be expressed as follows:

$$\boldsymbol{\Xi} = \begin{bmatrix} \frac{l}{2}\Lambda_T & l\Lambda_T & \frac{l}{2}\Lambda_T & -\frac{l}{2}\Lambda_T & -l\Lambda_T & -\frac{l}{2}\Lambda_T \\ -\frac{\sqrt{3}}{2}l\Lambda_T & 0 & \frac{\sqrt{3}}{2}l\Lambda_T & \frac{\sqrt{3}}{2}l\Lambda_T & 0 & -\frac{\sqrt{3}}{2}l\Lambda_T \\ \Lambda_C & -\Lambda_C & \Lambda_C & -\Lambda_C & \Lambda_C & -\Lambda_C \end{bmatrix} \tag{14}$$

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 $\dot{\boldsymbol{v}}$ and angular velocities $\boldsymbol{\omega}$ 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:

$$\boldsymbol{u} = \boldsymbol{u}^* + \boldsymbol{S}(\boldsymbol{\omega})\boldsymbol{J}\boldsymbol{\omega} + m\boldsymbol{c} \tag{15}$$

The linearized dynamic model can then be given as:

$$\boldsymbol{J}\dot{\boldsymbol{\omega}} = \boldsymbol{u}^* + \boldsymbol{d} \tag{16}$$

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:

$$J_i\dot{\omega}_i = u_i^* + d_i \tag{17}$$

By importing a new extended state vector $\boldsymbol{\zeta_i} = [\zeta_{i,1}, \zeta_{i,2}]^T$, 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:

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

where χ represents the derivative of d_i and it is assumed that $|\chi| < \nu^+$, 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):

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

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:

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

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

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

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

$$V_i = \boldsymbol{\delta_i}^T\boldsymbol{\eta_1}\boldsymbol{\delta_i} \tag{22}$$

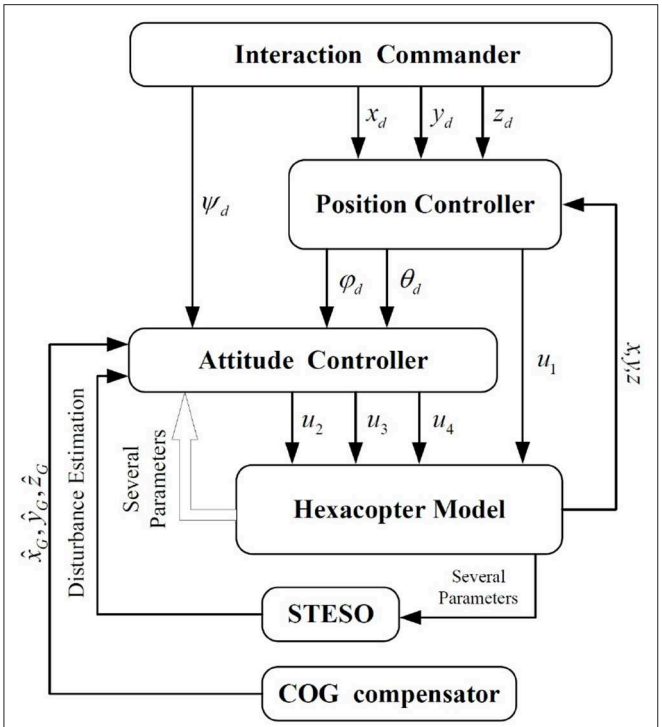

FIGURE 3 | Control schematic of the overall hexacopter UAV system.

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:

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

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:

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

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

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

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

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

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}:

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

By introducing a virtual control $\boldsymbol{\omega_{ed}} = -\mathrm{M}_1\boldsymbol{q_{ev}}$, 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:

$$\dot{V}_{A1} = -\boldsymbol{q_{ev}}^T\mathrm{M}_1\boldsymbol{q_{ev}} \leq 0 \tag{30}$$

$$\tilde{\boldsymbol{\omega}}_{\boldsymbol{e}} = \boldsymbol{\omega_e} + \mathrm{M}_1\boldsymbol{q_{ev}} \tag{31}$$

We then choose the Lyapunov function as:

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

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

$$\dot{V}_{A2} = \boldsymbol{q_{ev}}^T\tilde{\boldsymbol{\omega}}_e - \boldsymbol{q_{ev}}^T\boldsymbol{M_1}\boldsymbol{q_{ev}} + \tilde{\boldsymbol{\omega}}_e^T(\boldsymbol{J}\dot{\boldsymbol{\omega}}_e + \boldsymbol{J}\boldsymbol{M_1}\dot{\boldsymbol{q}}_{ev}) + \dot{V}_x + \dot{V}_y + \dot{V}_z = -\boldsymbol{q_{ev}}^T\boldsymbol{M_1}\boldsymbol{q_{ev}} + \tilde{\boldsymbol{\omega}}_e^T\left(\boldsymbol{J}(\boldsymbol{S}(\boldsymbol{\omega}_e)\boldsymbol{C}_d^b\boldsymbol{\omega}_d - \boldsymbol{C}_d^b\dot{\boldsymbol{\omega}}_d) - \boldsymbol{S}(\boldsymbol{\omega})\boldsymbol{J}\boldsymbol{\omega} + \boldsymbol{u} - m\boldsymbol{c} + \boldsymbol{d} + \boldsymbol{J}\boldsymbol{M_1}\dot{\boldsymbol{q}}_{ev} + \boldsymbol{q}_{ev}\right) + \dot{V}_x + \dot{V}_y + \dot{V}_z \tag{33}$$

By introducing the control input vector $\boldsymbol{u}$:

$$\boldsymbol{u} = -\boldsymbol{J}(\boldsymbol{S}(\boldsymbol{\omega}_e)\boldsymbol{C}_d^b\boldsymbol{\omega}_d - \boldsymbol{C}_d^b\dot{\boldsymbol{\omega}}_d) + \boldsymbol{S}(\boldsymbol{\omega})\boldsymbol{J}\boldsymbol{\omega} + m\boldsymbol{c} - \boldsymbol{J}\boldsymbol{M_1}\dot{\boldsymbol{q}}_{ev} - \boldsymbol{q}_{ev} - \boldsymbol{M_2}\tilde{\boldsymbol{\omega}}_e - \hat{\boldsymbol{d}} \tag{34}$$

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

$$\begin{aligned}\dot{V}_{A2} &= -\boldsymbol{q_{ev}}^T\boldsymbol{M_1}\boldsymbol{q_{ev}} - \tilde{\boldsymbol{\omega}}_e^T\boldsymbol{M_2}\tilde{\boldsymbol{\omega}} + \tilde{\boldsymbol{\omega}}_e^T\tilde{\boldsymbol{d}} + \dot{V}_x + \dot{V}_y + \dot{V}_z \\ &\le -\lambda_{\min}(\boldsymbol{M_1})\left\|\boldsymbol{q_{ev}}\right\|^2 - \lambda_{\min}(\boldsymbol{M_2})\left\|\tilde{\boldsymbol{\omega}}_e\right\|^2 + \left\|\tilde{\boldsymbol{\omega}}_e\right\|\left\|\tilde{\boldsymbol{d}}\right\| \\ &\quad -\lambda_{\min}(\boldsymbol{\eta_1})(\|\boldsymbol{\delta_x}\|^2 + \left\|\boldsymbol{\delta_y}\right\|^2 + \|\boldsymbol{\delta_z}\|^2) \\ &\le -\lambda_{\min}(\boldsymbol{M_1})\left\|\boldsymbol{q_{ev}}\right\|^2 - \lambda_{\min}(\boldsymbol{M_2})\left\|\tilde{\boldsymbol{\omega}}_e\right\|^2 + \left\|\tilde{\boldsymbol{\omega}}_e\right\|\|\boldsymbol{\delta}\| \\ &\quad -\lambda_{\min}(\boldsymbol{\eta_1})(\|\boldsymbol{\delta_x}\|^2 + \left\|\boldsymbol{\delta_y}\right\|^2 + \|\boldsymbol{\delta_z}\|^2) \\ &\le -\lambda_{\min}(\boldsymbol{M_1})\left\|\boldsymbol{q_{ev}}\right\|^2 - (\lambda_{\min}(\boldsymbol{M_2}) - \frac{1}{2})\left\|\tilde{\boldsymbol{\omega}}_e\right\|^2 \\ &\quad -(\lambda_{\min}(\boldsymbol{\eta_1}) - \frac{1}{2})\|\boldsymbol{\delta}\|^2\end{aligned} \tag{35}$$

where $\left\|\tilde{\boldsymbol{d}}\right\| = \|\boldsymbol{\delta_2}\| \le \|\boldsymbol{\delta}\|$, $\left\|\tilde{\boldsymbol{\omega}}_e\right\|\|\boldsymbol{\delta}\| \le \frac{1}{2}(\left\|\tilde{\boldsymbol{\omega}}_e\right\|^2 + \|\boldsymbol{\delta}\|^2)$, $\|\boldsymbol{\delta}\|^2 = \|\boldsymbol{\delta_x}\|^2 + \left\|\boldsymbol{\delta_y}\right\|^2 + \|\boldsymbol{\delta_z}\|^2$, and $\lambda_{\min}(\boldsymbol{M})$ denotes the minimal eigenvalue of $\boldsymbol{M}$. Thus, $\dot{V}_{A2} \le 0$ whenever $\lambda_{\min}(\boldsymbol{\eta_1}), \lambda_{\min}(\boldsymbol{M_2}) \ge \frac{1}{2}$. In that case, it can be concluded that the attitude error $\boldsymbol{q_e}$, angular velocity tracking error $\boldsymbol{\omega_e}$, and estimation errors $\boldsymbol{\delta_x}$,$\boldsymbol{\delta_y}$,$\boldsymbol{\delta_z}$ 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 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**.

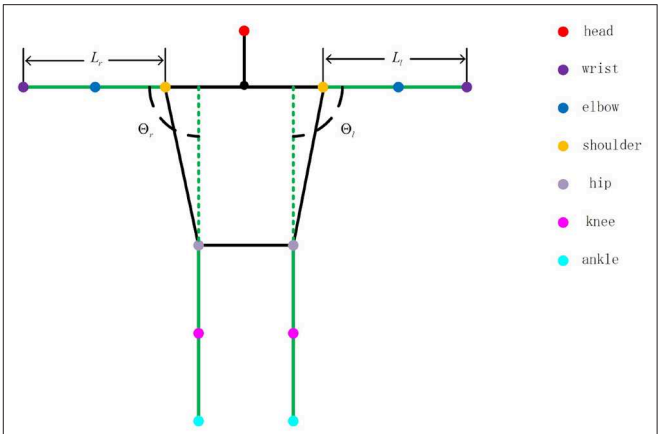

FIGURE 4 | Prototype of human joint.

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
$100 < \Theta_r < 150$ 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

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

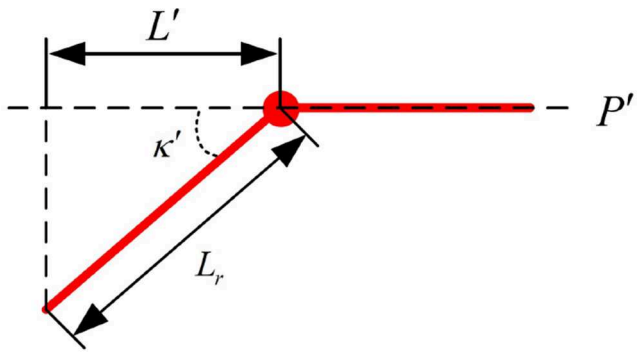

FIGURE 5 | Aerial view of nonobjective human.

Algorithm 1: The Whole Interaction Procedure.

```
Execute human pose estimation system initially and record video from camera;
if Detect interaction initiation action for more than 5 s then
    Measure depth information of the detected human and approach him immediately;
    Continue to execute human pose estimation system;
    if interaction initiation then
        if Detect flight direction command then
            Record target flight direction;
            if Detect flight distance command then
                Record target flight distance;
                if End flag then
                    Execute flight command from human and return to the first line in the procedure;
                end if
            end if
        end if
    end if
else
    Continue execute human pose estimation system and record the video from camera;
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:

$$\kappa' = \arccos \frac{L'}{L_r} \tag{36}$$

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:

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

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 $\boldsymbol{M_1} = 10\boldsymbol{I_3}$, $\boldsymbol{M_2} = 3\boldsymbol{I_3}$, the STESO gains as $\xi_{1,i} = 28$, $\xi_{2,i} = 58$, and the traditional second-order ESO gains as $\boldsymbol{L} = [\,35\ \ 380\ \ 600\,]^T$.

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

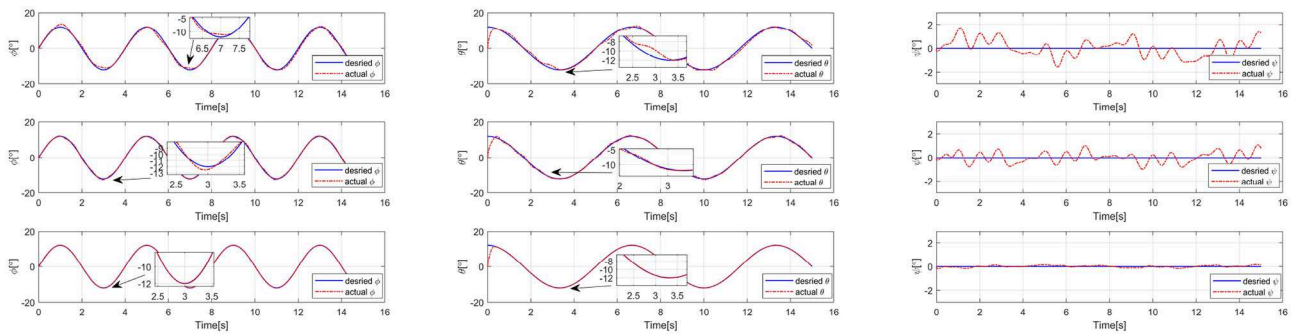

FIGURE 6 | Simulation results of roll, pitch, and yaw angle-tracking with unknown external torque disturbance. Without observer (Up), traditional ESO (Middle), and STESO (Down).

simulations have been conducted based on the proposed back-stepping controller with the STESO and traditional ESO. The desired attitude references are given as:

$$\boldsymbol{\Theta_d} = [12\sin(0.5\pi t) \quad 12\cos(0.3\pi t) \quad 0\ ]^T \ \text{deg} \tag{38}$$

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.

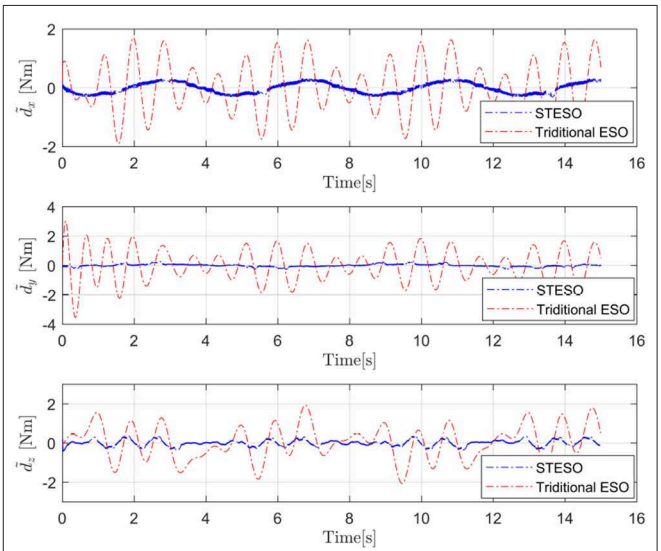

FIGURE 7 | Disturbances estimate errors ($\tilde{d}_x, \tilde{d}_y, \tilde{d}_z$) of the STESO and traditional ESO.

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, 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

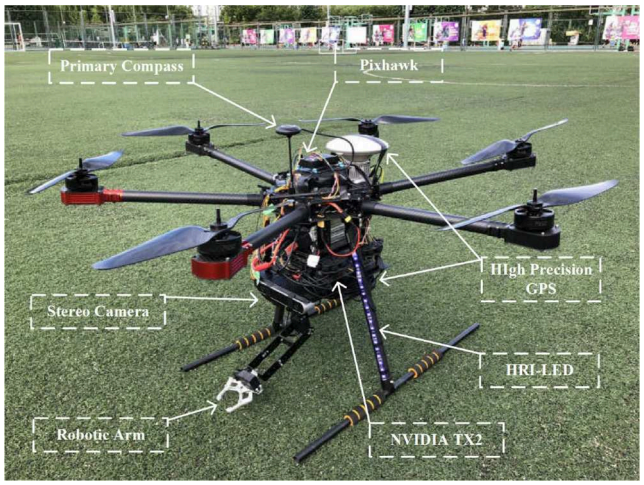

FIGURE 8 | Prototype of the proposed interactive UAV.

FIGURE 10 | Interaction between human and UAV. Written informed consent for publication was obtained from the individuals in this image.

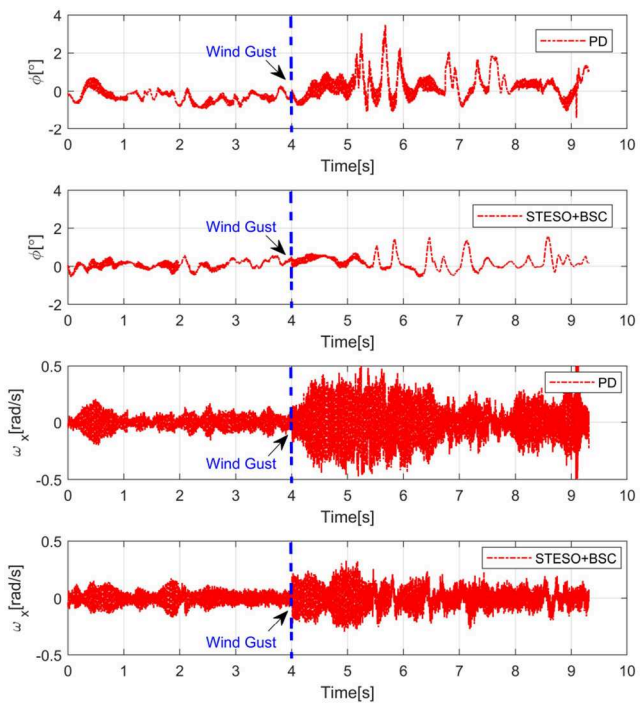

FIGURE 9 | Experimental curves of attitude angle φ and angular rate ω_x during hovering with wind gusts.

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

model uncertainties, a hovering experiment was conducted with wind disturbance generated by several electrical fans. We set the BSC gains at $\boldsymbol{M_1} = diag(10, 10, 4)$ and $\boldsymbol{M_2} = diag(0.2, 0.2, 0.28)$ 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 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

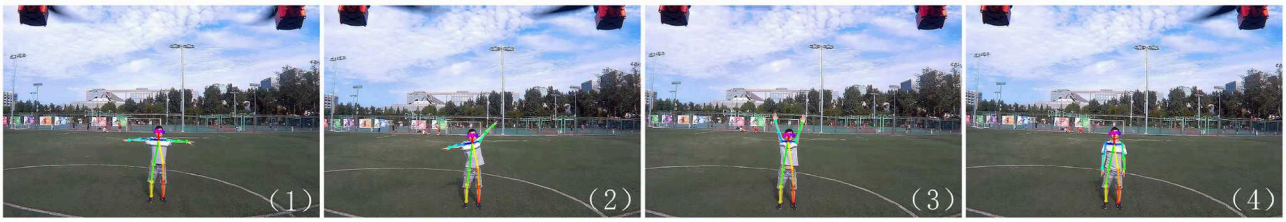

FIGURE 12 | Target direction and distance communication to the UAV. (1) Beginning. (2) Target direction command. (3) Target distance command. (4) End. Written informed consent for publication was obtained from the individual in this image.

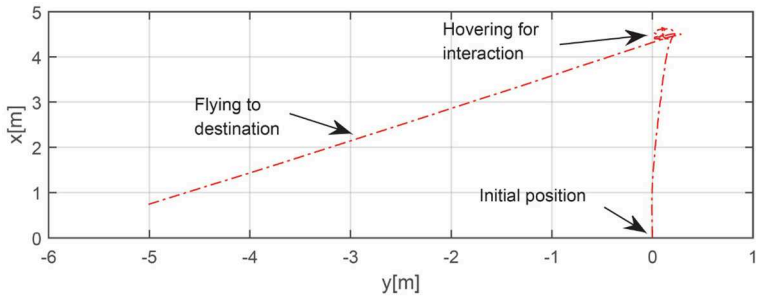

FIGURE 13 | Trajectory in automated exploration task.

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

This work was supported by the National Key R&D Program of China (Grant No. 2019YFB1310802) and the National Natural Science Foundation of China (Grant No. 61633002).

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

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.