frontiers | Frontiers in Computational Neuroscience

TYPE Original Research
PUBLISHED 20 February 2023
DOI 10.3389/fncom.2023.1110889

OPEN ACCESS

EDITED BY
Wujie Zhou,
Zhejiang University of Science and Technology, China

REVIEWED BY
Arpit Sharma,
Manipal University Jaipur, India
Manoj Kumar,
University of Wollongong in Dubai, United Arab Emirates

*CORRESPONDENCE
Yuan Li
✉ sczyxxzx@163.com

RECEIVED 29 November 2022
ACCEPTED 16 January 2023
PUBLISHED 20 February 2023

CITATION
Zhang Y, Xie L, Li Y and Li Y (2023) A neural learning approach for simultaneous object detection and grasp detection in cluttered scenes. *Front. Comput. Neurosci.* 17:1110889. doi: 10.3389/fncom.2023.1110889

A neural learning approach for simultaneous object detection and grasp detection in cluttered scenes

Yang Zhang[1], Lihua Xie[1], Yuheng Li[2] and Yuan Li[1]*

[1]China Tobacco Sichuan Industrial Co., Ltd, Chengdu, Sichuan, China, [2]Qinhuangdao Tobacco Machinery Co., Ltd, Qinhuangdao, Hebei, China

Object detection and grasp detection are essential for unmanned systems working in cluttered real-world environments. Detecting grasp configurations for each object in the scene would enable reasoning manipulations. However, finding the relationships between objects and grasp configurations is still a challenging problem. To achieve this, we propose a novel neural learning approach, namely SOGD, to predict a best grasp configuration for each detected objects from an RGB-D image. The cluttered background is first filtered out via a 3D-plane-based approach. Then two separate branches are designed to detect objects and grasp candidates, respectively. The relationship between object proposals and grasp candidates are learned by an additional alignment module. A series of experiments are conducted on two public datasets (Cornell Grasp Dataset and Jacquard Dataset) and the results demonstrate the superior performance of our SOGD against SOTA methods in predicting reasonable grasp configurations "from a cluttered scene."

KEYWORDS

grasp detection, object detection, RGB-D image, deep neural network, robotic manipulation

1. Introduction

Automated object grasping is essential and challenging to robots or unmanned systems working in real-world cluttered scenarios. As a core component of autonomous grasping, grasp detection, which outputs the most possible grasp configuration for the manipulator, has attracted great attention from both academic and industrial communities. Existing methods often predict a series of possible grasp configurations based on the input images (Depierre et al., 2018; Zhang et al., 2019; Wang et al., 2021; Yu et al., 2022b). When encountered with a cluttered scene, which is a common case in our daily life, we humans often identify the target object first and then determine the best pose to grab the object. This provides two kinds of benefits: (1) we can easily explain why the predicted grasp configuration is the best, and (2) our efforts will be focused on the object area instead of the cluttered background to make a better decision. However, most previous studies do not have a strong ability to model the relationship between the target objects and the predicted grasp configurations. In order to make grasp detection more accurate and reasonable, we investigate the problem of simultaneous object detection and grasp detection, where the best grasp configuration is predicted for each detected object in the cluttered scene.

Since object manipulation is performed in a 3D space, using a 3D representation for grasp detection is a natural way. A grasp candidate is a 6-DOF gripper pose $g = (x, y, z, r_x, r_y, r_z)$, $g \in SE(3)$, with the 3D position and rotation angles along each axis of the gripper. Methods based on this 3D representation (Pas et al., 2017; Liang et al., 2019) often generate a large number of candidates and then evaluate whether it is a good grasp according to a specific criterion. These methods are easy to understand but often suffer efficiency problems due to 3D operations. Motivated by the superior performance of deep learning technology on detection

or segmentation tasks (Cheon et al., 2022; Huang et al., 2022; Khan et al., 2022), image-based deep models have become popular for grasp detection (Chu et al., 2018; Zhang et al., 2019; Dong et al., 2021; Yu et al., 2022a). These methods often use a rectangle representation $g = (x, y, h, w, \theta)$, where (x, y) is the center pixel location of a grasp candidate, (h, w) are height and width of the gripper, and θ is the rotation of the gripper. This representation is widely used in end-to-end deep networks. Some other studies (Wang et al., 2019, 2022) also used a score map with the same size as the image to represent the quality of grasp configurations at each pixel.

A number of existing grasp detection methods are inspired by object detection (Zhou et al., 2018; Zhang et al., 2019; Park et al., 2020). These two problems share a similar output, which consists of a regression of a rectangle (a grasp configuration for grasp detection or a bounding box for object detection) and a classification score (quality of the grasp or confidence in the predicted category). Thus, one straightforward way for designing a grasp detection model is to modify it from an object detector. For example, Zhang et al. (2019) propose an ROI-based grasp detection method which is a modification from Faster R-CNN. They use the region proposal network (RPN) to generate graspable proposals and an ROI-pooling layer to extract features for each proposal. Then grasp configurations and corresponding successful rates are estimated with the local features. However, grasp detection differs from object detection in the additional prediction of orientation. To predict the orientation of the gripper, serval existing methods (Chu et al., 2018; Dong et al., 2021; Yu et al., 2022a) convert this regression problem into a classification problem by discretizing continuous angles into angle anchors. This makes orientation prediction much more convenient but will also cause a loss of accuracy. To overcome this short back, other studies (Park et al., 2020) use classification and regression processes to predict the final angles. Another kind of grasp detection method (Yu et al., 2022b) makes dense predictions at each pixel and outputs a set of heatmaps representing the grasp configurations and quality.

To generate more reasoning and human-like grasp candidates, we investigate the problem of simultaneously detecting objects and grasp configurations from an RGB-D image and propose a novel neural learning approach, namely SOGD, for this task. Our SOGD model takes an RGB-D image as input and outputs a set of tuples $(x_o, y_o, w_o, h_o, cls_o, x_g, y_g, w_g, h_g, \theta_g, s_g)$, representing the joint prediction of the object detection result and the grasp detection result. To this end, features extracted by the top stages of a backbone and feature pyramid network (FPN) are used for both detection tasks. Two separate detection branches are designed to detect objects and grasp them, respectively. The correspondences between object proposals and grasp candidates are modeled by an alignment module. In addition, we present a depth-based method to filter out backgrounds in a cluttered scene. This would enable our detectors to focus on features from the target objects other than the texture from the environment.

Our main contributions are summarized as follows:

(1) We propose a novel neural learning approach to detect target objects and their best grasp configurations in cluttered environments simultaneously.
(2) An alignment module is designed to estimate the correlations between the separately detected objects and grasp configurations. This module enables our model to predict more reasonable grasp configurations for each detected object.
(3) A 3D-plane-based pre-processing is presented to filter out cluttered backgrounds from the RGB-D image.
(4) A series of experiments are conducted on two publicly available datasets (the Cornell Grasp Dataset and the Jacquard Dataset). Our method achieves +0.7 to +1.4% improvement in average accuracy compared with the existing RGB-D-based grasp detection methods.

2. Related studies

Grasp detection methods can be divided into traditional methods and learning-based methods. The traditional methods are mainly divided into the template matching method and the feature point matching method. The template-based pose estimation algorithm (Georgakis et al., 2019) needs to build the template of the object in advance, which can be strongly applicable to objects with regular shapes and has a good effect on targets without texture. However, when the object is blocked and the light is insufficient, the matching will be too low, leading to the failure of prediction. The pose estimation method based on feature points can extract effective feature points from images and match them with standard images. Since descriptors can describe visual features stably and robustly, this method is not susceptible to illumination. However, this method only uses the information of feature points in the image, so the utilization rate of information is very low. If there are not many feature points in the image, this method will have a high probability of deviation from the predicted capture rectangle.

Motivated by the superior performance of deep learning technology (Chhabra et al., 2022; Motwani et al., 2022; Shailendra et al., 2022; Singh et al., 2022), it has been applied in grasping detection to improve the accuracy of grasping in recent years. In order to improve the generalization of 3D models, some grab detection methods based on 3D reconstruction are proposed. According to Yang et al. (2021), this method uses 3D reconstruction to optimize the candidate grasping objects generated by the grasping suggestion network and improves the grasping accuracy of unknown objects. According to Jiang et al. (2021), this method uses implicit neural representation and studies synergies between affordance and geometry to improve the accuracy of grasping detection. Sundermeyer et al. (2021) used a 3D point cloud to predict the 3D points of grasping contact and reduce the dimension from 6-Dof to 4-Dof to make the learning process more convenient. However, the method based on 3D reconstruction needs a certain amount of time to build the 3D model, and the real-time capturing will be affected to some extent.

Since deep learning has shown excellent results in detection and segmentation tasks, image-based capture detection methods have become increasingly popular. But different from object detection, grasp detection needs to predict the angle of the gripper. Therefore, some of the methods (Chu et al., 2018; Dong et al., 2021; Yu et al., 2022a) discretized continuous angles into angle anchors and transformed the regression problem into a classification problem. However, these methods may cause a lack of accuracy. To solve

this problem, Park et al. (2020) predicted the final angle using classification and regression processes. The method provided by Yu et al. (2022b) intensively predicts that each pixel represents the heat map of the capture configuration and quality. Zhang et al. (2018) divided the grasping problem into two separate tasks (object detection and grab detection) and then integrated them as the final solution. Yu et al. (2022b) proposed a module that extracts feature mappings from bidirectional feature pyramid networks, object detection, and grab detection, and outputs the optimal grab position and appropriate operational relations. Park et al. (2020) predicted the boundary box, the category of objects, and the direction of the grab rectangle and grab configuration using a global feature map. Ainetter and Fraundorfer (2021) designed an end-to-end CNN-based network architecture and designed a refinement module to improve the accuracy of prediction.

Most deep-learning-based methods directly output grasp candidates without recognizing the target object. They cannot answer the question that what is the best grasp configuration for every single object in a cluttered scene. Unfortunately, this is a common case an unmanned system needs to deal with. To generate a more reasoning prediction, Zhang et al. (2018) designed a recognize-and-then-grasp approach, which divides the problem into two separate tasks (object detection and grasp detection) and then integrates them as the final solution. Another way is to perform grasp detection together with object detection or segmentation tasks (Park et al., 2020; Ainetter and Fraundorfer, 2021; Yu et al., 2022a). For example, Park et al. (2020) generated a global feature map to predict the bounding box, the category of an object, the grasp rectangle, and the orientation of a grasp configuration. Then, non-maximum suppression is applied to both bounding boxes and grasp rectangles to filter out unnecessary predictions. The relationship between the bounding boxes and the grasp rectangles is built *via* computing the Intersection over Union (IoU) of the two areas. If the IoU is greater than a certain threshold, the grasp will be assigned to the detected object. However, choosing an appropriate threshold is not easy. When the graspable area is much smaller than the entire object, the IoU between the grasp rectangle and the bounding box of the object will be too small. As a result, such a strategy cannot avoid filtering out possible solutions.

3. Materials and methods

3.1. Problem formulation and reparameterization

In the process of object grabbing, humans usually first identify the object to be grabbed in the scene and then select an appropriate grabbing position for the target object. Whether a grasping pose is appropriate is directly related to the target object to be grasped. Motivated by this fact, this study investigates the problem of robotic manipulation by detecting the target object in the scene and its grasp position simultaneously.

Given an RGB-D image, the goal of the simultaneous object and grasp detection is to identify every single object in the scene and find out a grasp configuration for it. To this end, we formulate the representation of the simultaneous object and grasp detection det as:

$$\text{det} = (\mathbf{od}, \mathbf{gd})$$

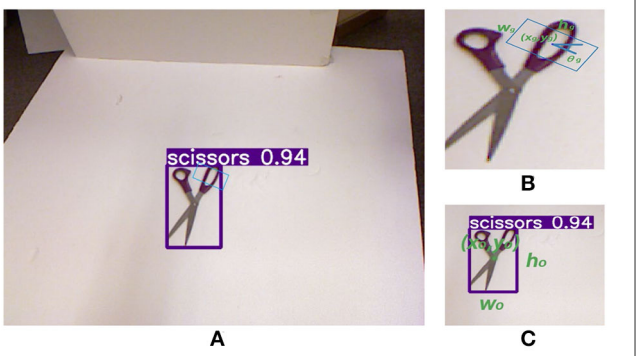

FIGURE 1
An example of our simultaneous object and grasp detection representation. **(A)** 11D object and grasp detection representation. **(B)** 5D object detection representation with location (x_o, y_o), width w_o, height h_o, and the category of the object cls_o. **(C)** 6D grasp detection representation with location (x_g, y_g), gripper width w_g, plate size h_g, orientation θ_g, and its success rate s_g.

$$\mathbf{od} = (x_o, y_o, w_o, h_o, cls_o)$$

$$\mathbf{gd} = (x_g, y_g, w_g, h_g, \theta_g, s_g)$$

The representation consists of two parts: object detection and grasp detection. Figure 1 shows an example of this representation. For the object detection part, we use (x_o, y_o, w_o, h_o) to represent the location of a bounding box and cls_o to represent the category of the object inside of it. For the grasp detection part, we adopt the famous 5-dimensional rectangular representation (Lenz et al., 2013), which consists of the location and orientation $(x_g, y_g, w_g, h_g, \theta_g)$ of the gripper for a grasp configuration. In addition, we add s_g, a value between 0 and 1, to represent the score of a grasp. The higher s_g is, the greater chance of the grasp being a success.

Similar to Park et al. (2020), we formulate the estimation of θ_g as a classification + regression problem instead of a single regression problem. According to the symmetry of the gripper, the range of θ_g is $[0, \pi]$. We convert this range into several bins $\left\{0, \frac{\pi}{k_a}, \frac{2\pi}{k_a}, ..., (k_a - 1)\frac{\pi}{k_a}\right\}$ to be angle anchors, with k_a is the number of bins. The classification problem is to predict a one-vs.-all vector to determine which bins θ_g belongs to. The regression problem is to estimate the angle offset to the anchors.

Inspired by Redmon and Farhadi (2018) and Ge et al. (2021), we reparametrize the regression problem of $(x_j, y_j, w_j, h_j, \theta_j)$ as estimation of $(t_j^x, t_j^y, t_j^w, t_j^h, t_j^\theta)$ to the location of the grid cell (a_j^x, a_j^y), bounding box prior width and height (a_j^w, a_j^h), and orientation angle bin a_j^θ. The relationship between $(x_j, y_j, w_j, h_j, \theta_j)$ and $(t_j^x, t_j^y, t_j^w, t_j^h, t_j^\theta)$ is defined as follows.

$$x_j = \sigma(t_j^x) + a_j^x$$

$$y_j = \sigma(t_j^y) + a_j^y$$

$$w_j = a_j^w \times e^{t_j^w}$$

$$h_j = a_j^h \times e^{t_j^h}$$

$$\theta_j = \sigma(t_j^\theta) \times \left(\frac{\pi}{k_a}\right) + a_j^\theta$$

This reparameterization is applied to both the object bounding box regression and the grasp rectangle regression.

3.2. Overview of the SOGD model

The architecture of our SOGD model is shown in Figure 2. It takes an RGB-D image as input, and outputs the detected object's bounding $(t_o^x, t_o^y, t_o^w, t_o^h)$ and category *cls* together with the corresponding grasp position $(t_g^x, t_g^y, t_g^w, t_g^h)$, orientation t_g^θ, and success rate s_g. Our model consists of five parts: a pre-processing for background removal, a backbone and a feature pyramid network (FPN) for image feature extraction, an object detection head, a grasp detection head, and an alignment module for candidate objects and grasp configurations.

Motivated by Dong et al. (2021), we design a pre-processing module to remove the background from the cluttered scene. According to Dong et al. (2021), backgrounds are recognized by an encoder–decoder network to segment the original image. However, our module utilizes the priors of the scene that objects to be grabbed are laid on a 3D plane, such as the surface of a desk. According to this fact, we categorize pixels on and under the 3D plane as background and filter them out. We argue that this background removal strategy is more reasonable and robust than the U-net-based method (Dong et al., 2021). Details about the 3D-plane-based background removal are discussed later.

For multi-scale feature extraction, various deep models [e.g., Darknet (Wood, 2009) or ResNet (He et al., 2016)] can be utilized. In this study, ResNet-50 is used as the backbone to release the computational burden of deep models during feature extraction and facilitate real-time performance. The very last feature map learned by different stages (e.g., *conv1*, *conv2*, and *conv3*) is used as multi-scale features. We denote these feature maps as $\{C_1, C_2, C_3, C_4, C_5\}$. The stride steps of these feature maps are $\{2, 4, 8, 16, 32\}$ with respect to the original image. Only $\{C_3, C_4, C_5\}$ are used in FPN for feature fusion and the fused feature maps are denoted as $\{P_3, P_4, P_5\}$. Our feature extraction and fusion module can be formulated as follows:

$$image\_f = BackgroundRemoval\left(image\_rgbd\right)$$

$$F_{backbone} = \{C_3, C_4, C_5\} = ResNet\left(image\_f\right)$$

$$F_{FPN} = \{P_3, P_4, P_5\} = FPN\left(F_{backbone}\right)$$

Inspired by the fact that grasp rectangles are often much small than the object's boundaries, we use different prior rectangle sizes for object detection and grasp detection. The object detection head and the grasp detection head share a similar structure with the detection head (Redmon and Farhadi, 2018). The output tensor for object detection is in the shape of $N \times N \times k_o \times (4 + 1 + C_o)$, where $N \times N$ is the size of the feature map, k_o is the number of predicted bounding boxes, 4 stands for the number of parameters of a bounding box, 1 stands for the channel of confidence, and C_o is the number of object categories. Similarly, the output tensor for grasp detection is in the shape of $N \times N \times k_g \times (5 + C_a + 1)$, where 5 stands for the location, width, height, and orientation of a grasp rectangle, C_a is the number of angle bins, and 1 stand for the successful rate prediction.

3.3. Background removal

Robotic manipulation often encounters a cluttered environment. The captured RGB-D images include both the target objects to be grasped and the background surroundings. To achieve an accurate object and grasp detection, we should focus on the pixels belonging to the targets. The additional observation on backgrounds may distract our attention from the targets. Figure 3 presents a quantitative analysis of this additional information. When the background is removed, the detection model only needs to learn features from the target objects and all learned features are valuable for final manipulation. However, if the RGB-D image encounters a cluttered background, the model will have to learn features from both the targets and the background, and distinguish which feature contributes to the downstream tasks. This will increase the burden of the model for feature extraction and also increase the number of parameters to learn task-specific features. According to Dong et al. (2021), these additional cluttered background pixels will even lead to a false grasp detection.

To eliminate the cluttered background and let the model focus on the targets, Dong et al. (2021) adopt an encoder–decoder-based U-net model to segment the input image into foreground and background. It is indeed a potential way to filter out the background in an image. But the U-net model needs to be trained on a large dataset and its generalizability to new observations is limited. Instead of recognizing the background in the image domain, we present a background removal method in 3D space. We assume that objects to be grasped are laid on a 3D plane (such as the surface of a desk), which is the common case in robotic manipulation. Under this assumption, pixels that are up to the 3D plane are defined as the foreground, while pixels in or under the 3D plane are defined as the background. This could separate the targets from the cluttered background in most cases. In the top-left image in Figure 3, the white vertical surface will also be considered as the foreground using the aforementioned approach. But this mis-segmentation will not affect the detection of the targets since the vertical surface is disconnected from the targets.

For the 3D plane estimation, we use a model-based method to fit the unknown parameters. In 3D spaces, a plane is defined as $aX + bY + cZ + d = 0$, with (a, b, c, d) as the plane parameters. Given three points, we can fit a plane for it. Since pixels belonging to the 3D plane are dominant in the image, we adopt a RANSAC-based method to fit the parameters of the largest 3D plane in the image. The method achieves its goal by iteratively selecting a random subset of the original 3D points. The selected subset is assumed to be inliers and the plane parameters are fitted with respect to these inlier points. Then all other points are tested against the fitted model. If a point fits well to the estimated model, it will be considered as inliers to the model. The fitted 3D plane is reasonably good if sufficiently many points are classified as inliers. In this study, 3D coordinates are computed from the depth image with a fixed camera intrinsic parameter when 3D points are not presented in the dataset.

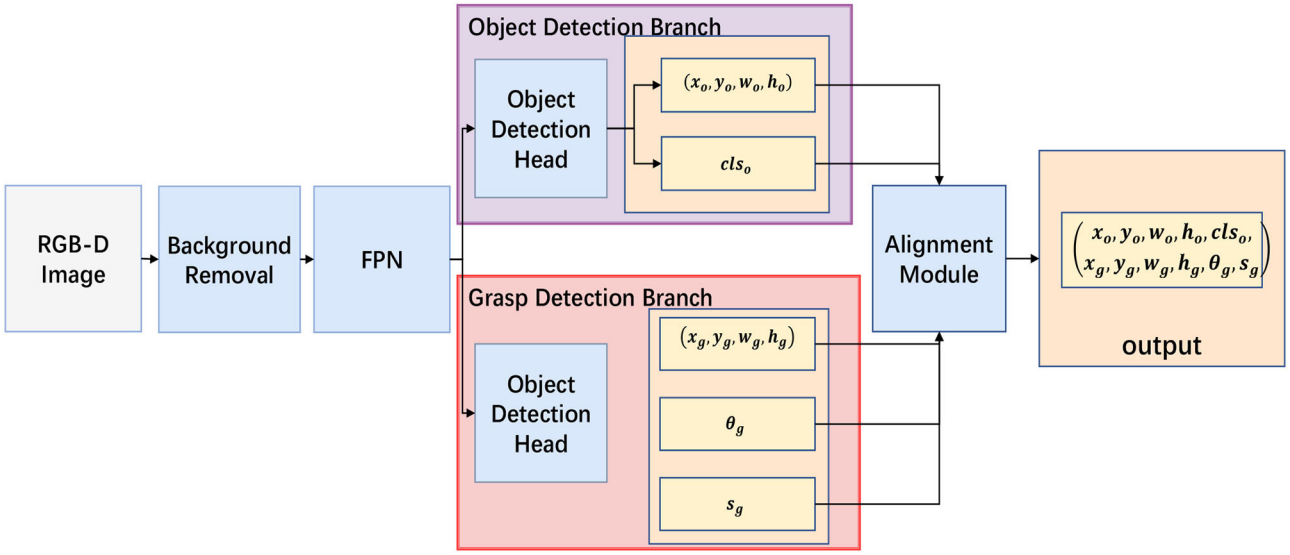

FIGURE 2
Outline of our Simultaneous Object and Grasp Detection (SOGD) model. SOGD takes an RGB-D image as input and outputs a series of recognized objects together with the most appropriate grasp for every single object. It consists of five parts: a pre-processing module to remove background from the cluttered scene, a backbone (e.g., Darknet or ResNet) and FPN for hierarchical feature extraction, two separate branches for object detection and grasp detection, and an alignment module to assign a most appropriate grasp for each detected object.

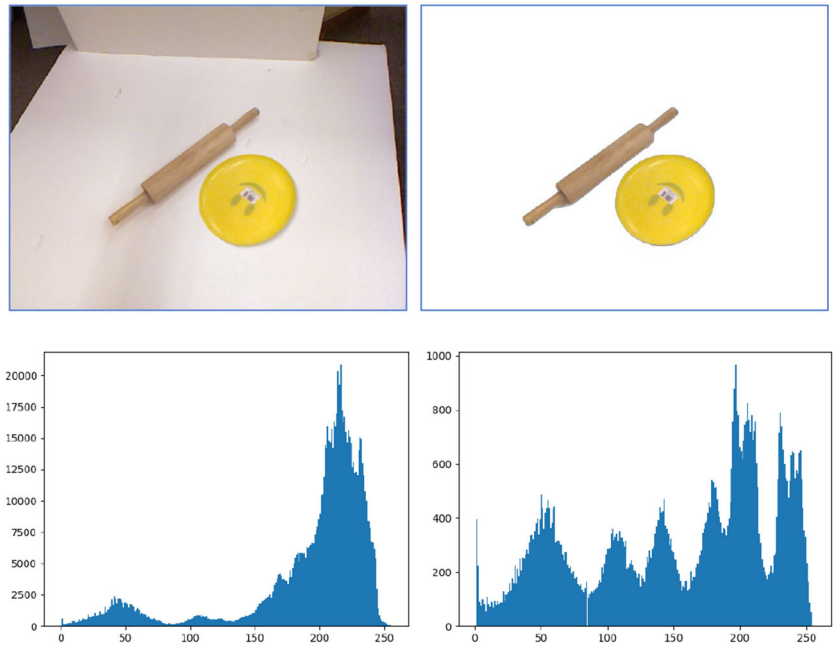

FIGURE 3
Illustration of the unnecessary efforts spent on the cluttered background during feature extraction. **Top** lines are original images and foreground images. **Bottom** lines are the corresponding histograms of the top images. From the histograms, it can be seen that the additional information which is useless to target detection will significantly increase when a cluttered background is encountered in the image.

We also applied voxelization to speed up the process and generate a finer plane.

3.4. Separate object and grasp detection branches

Object detection and grasp detection are both detection problems. These two tasks share a similar output in the regression of location, width and height of a rectangle (as the bounding box for object detection and the grasp rectangle for grasp detection), and a confidence score (as the classification for object detection and the grasp quality for grasp detection). According to observation, we design two separate detection branches for these two tasks, but the branches share a similar architecture as shown in Figure 4.

The structure of our two detection branches is motivated by modern object detectors (Redmon and Farhadi, 2018). The detection head takes multi-stage outputs from FPN as inputs to detect objects

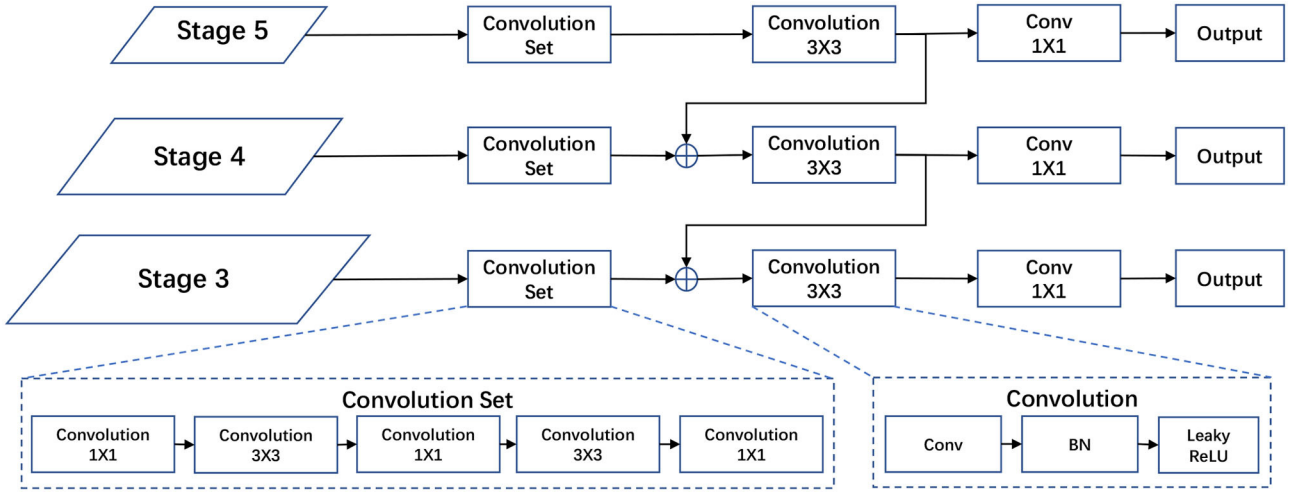

FIGURE 4
Structure of the object/grasp detection head. Our object detection head and grasp detection head share a similar structure except for the output channels. Both heads take $\{P_3, P_4, P_5\}$ from the FPN as inputs. The output of object detection includes the bounding box and object categories, while the output of grasp detection has more channels for orientation and grasp score.

or grasp configurations at different scales. In fact, a gripper only needs to grab a small part of an object to take it up, instead of grabbing the whole of it. As a result, the detected rectangle for grasp is often smaller than the bounding box of the object. Thus, we use the same inputs as modern detectors for object detection and grasp detection, but a relatively small-scale prior size for grasp detection. This enables our model to use a same-level semantic feature for both tasks.

Our detection head consists of a convolution set, a 3×3 convolution block, and a 1×1 convolution layer for prediction. The multi-stage outputs of FPN are treated as fused features which include both texture and semantic information extracted from the input image. The convolution set is designed to learn a task-correlated feature representation from the texture and semantic information. Then the 3×3 convolution block fuses task-corelated features at the top and current scales. The 1×1 convolution layer is used to match the number of channels to the final predictions. The number of channels of outputs for object detection is $k_o \times (4 + 1 + C_o)$, where k_o is the number of predicted bounding boxes for each grid cell; 4 stands for $(t_o^x, t_o^y, t_o^w, t_o^h)$, parameters of a bounding box; 1 stands for the confidence of the prediction; and C_o is the number of object categories. Similarly, the number of channels of output for grasp detection is $k_g \times (5 + C_a + 1)$, where k_g is the number of predicted grasp rectangles; 5 stands for $(t_g^x, t_g^y, t_g^w, t_g^h, t_g^\theta)$, parameters of a grasp rectangle; 1 stands for the score of the grasp configuration; and C_a is the number of angle bins. The mathematical computation of our detection head is as follows:

$$\mathbf{F_{task}} = \{\mathbf{T_3}, \mathbf{T_4}, \mathbf{T_5}\} = \mathbf{ConvolutionSet}\,(F_{FPN})$$

$$\mathbf{F_{task\_fusion}} = \{\mathbf{TF_3}, \mathbf{TF_4}, \mathbf{TF_5}\}$$

$$\mathbf{TF_i} = \mathbf{Convolution}\,(\mathbf{TF_{i+1}} + \mathbf{T_i})$$

$$\mathbf{Pro} = \mathbf{Conv_{1\times1}}\left(\mathbf{F_{task\_fusion}}\right)$$

3.5. Alignment between objects and grasp configurations

The two detection branches make predictions for objects and grasp configurations separately. To model the correspondence between detected objects and grasp configurations, we design an alignment module. Given an object prediction $\mathbf{Pro_o} \in \mathbf{R}^{N\times N\times k_o}$ and a grasp prediction $\mathbf{Pro_g} \in \mathbf{R}^{N\times N\times k_g}$, the correspondences between all possible pairs are defined as $\mathbf{Pro_{corr}} \in \mathbf{R}^{(N\times N\times k_o)\times(N\times N\times k_g)}$. Objects and grasp configurations are detected at different scales. Generating correspondences across multi scales would significantly increase the computational complexity. Thus, we only consider possible object-grasp pairs within the same scale.

Our alignment module takes the task-correlated features from object detection head $\mathbf{TF_o} \in \mathbf{R}^{N\times N\times c_o}$ and grasp detection head $\mathbf{TF}_g \in \mathbf{R}^{N\times N\times c_g}$ as input. Then two 1×1 convolution layers are applied to the features separately, resulting in the outputs of $\acute{\mathbf{F}}_\mathbf{o} \in \mathbf{R}^{N\times N\times k_o\times f}$ and $\acute{\mathbf{F}}_g \in \mathbf{R}^{N\times N\times k_g\times f}$. The two feature maps are reshaped into a 2D matrix and transpose matrix multiplication is applied to generate the output $\mathbf{F_{corr}} \in \mathbf{R}^{(N\times N\times k_o)\times(N\times N\times k_g)}$. Finally, we use a sigmoid activation function to model the joint possibility of the detection pairs. The mathematical computation of our alignment module can be formulated as follows:

$$\acute{\mathbf{F}}_\mathbf{o} = \mathbf{Conv_{1\times1}}\,(\mathbf{TF_o})$$

$$\acute{\mathbf{F}}_g = \mathbf{Conv_{1\times1}}\,(\mathbf{TF_g})$$

$$\mathbf{F}_{corr} = \mathbf{reshape}(\acute{\mathbf{F}}_\mathbf{o}) \bullet (\mathbf{reshape}(\acute{\mathbf{F}}_g))^T$$

$$\mathbf{Pro}_{corr} = sigmoid(\mathbf{F}_{corr})$$

Though our two detection heads make predictions separately, our model is forced to learn a better $\mathbf{Pro}_{corr}$ to model the correlation between the predictions. At inference, we use two additional parameters k_o and k_c to control the number of final predictions. First, top k_o object predictions are selected from $\mathbf{Pro}_o$, then top k_c correlated grasp predictions are selected and assigned to the detected objects. If the quality of a grasp prediction is smaller than a threshold, the grasp prediction will be filtered out from the alignment results. In this way, our model can be easily extended to multi-object detection and multi-grasp detection cases.

3.6. Loss function

The loss of our SOGD model consists of three parts: object detection loss L_o, grasp detection loss L_g, and alignment loss L_{corr}. The loss of object detection is defined as:

$$L_o = L_o^{reg} + \alpha \times L_o^{cls}$$

$$L_o^{reg} = \frac{1}{N_o^{reg}} \sum_i smooth_{L1}\left(t_o^i, \hat{t}_o^i\right)$$

$$L_o^{cls} = \frac{1}{N_o^{cls}} \sum_i Loss_{focal}\left(cls_i, \hat{cls}_i\right)$$

where t_o^i and $\hat{t}_o^i$ are ground truth and predictions for a bounding box, respectively. We use the smooth L1 loss for regression and focal loss (Lin et al., 2017) for classification. N_o^{reg} and N_o^{cls} are the normalizers. α is the weight of classification loss.

Similarly, the loss of grasp detection is defined as:

$$L_g = L_g^{reg} + \beta \times L_g^{angle} + \gamma \times L_g^{score}$$

$$L_g^{reg} = \frac{1}{N_g^{reg}} \sum_i smooth_{L1}\left(t_g^i, \hat{t}_g^i\right)$$

$$L_g^{angle} = -\sum_i \left[a_i \log\left(\hat{a}_i\right) + (1-a_i) \log\left(1-\hat{a}_i\right)\right]$$

$$L_g^{score} = -\sum_i \left[s_i \log\left(\hat{s}_i\right) + (1-s_i) \log\left(1-\hat{s}_i\right)\right]$$

where t_g^i and $\hat{t}_g^i$ are ground truth and predictions for a grasp rectangle, respectively. β and γ are hyperparameters.

The loss of object and grasp alignment is defined as:

$$L_{corr} = -\delta \times \sum_i \left[p_i \log\left(\hat{p}_i\right) + \left(1-p_i\right) \log\left(1-\hat{p}_i\right)\right]$$

where p_i and $\hat{p}_i$ are ground truth and predictions of a candidate pair for object and grasp, respectively. δ controls the weights of alignment loss to the total loss.

The total loss function is the summation of the three losses:

$$L = L_o + L_g + L_{corr}$$

4. Results

To evaluate the performance of our proposed SOGD model against previous methods, we test it on two publicly available datasets: the Cornell Grasp Dataset (Lenz et al., 2013) and the Jacquard Dataset (Depierre et al., 2018). Our model is designed for the task of grasp detection, but it also outputs predictions for object detection. Thus, the metrics used in our experiments consist of two parts. For the grasp detection task, we use the popular Jaccard Index and angle difference as metrics, consistent with previous methods (Jiang et al., 2011; Chu et al., 2018; Kumra et al., 2020; Yu et al., 2022b). A predicted grasp configuration is considered as correct if and only if the following two conditions are satisfied.

(1) Jaccard Index of the predicted grasp rectangle and the ground truth is >0.25. Assuming that $\hat{b}_g$ is the predicted grasp rectangle and b_g is the ground truth, Jaccard Index is defined as:

$$Jaccard\ Index = \frac{\left|\hat{b}_g \cap b_g\right|}{\left|\hat{b}_g \cup b_g\right|}$$

(2) the difference between the predicted orientation angle and the ground truth is <30°.

For object detection tasks, we use accuracy instead of the commonly used mAP as metrics. In this research, object detection is an additional output only to achieve the final goal of predicting the most possible grasp for each individual object in a cluttered scene. Our method only needs to know where the object is, and what kind of object it is does not matter too much. So, we consider a prediction as correct if the intersection over union of the predicted bounding box and ground truth is >0.5.

4.1. Grasp detection results on cornell grasp dataset

There are 878 images together with the corresponding depth image and 3D point clouds in the Cornell Grasp Dataset (Lenz et al., 2013). The resolution of the images is 640 × 480. Each image contains a single graspable object at different positions and orientations. The dataset is manually annotated with many positive and negative grasp rectangles. Following previous research (Zhang et al., 2019; Dong et al., 2021), we use a five-fold cross-validation strategy to evaluate the performance of our method and report the average detection accuracy in this section. The reported results include both image-wise (IW) and object-wise (OW) detection accuracy. In image-wise experiments, all images are randomly divided into a train set and a test set. The object in the test set may have been learned during training but at different poses and views. This is mainly to test the generalization ability of our method when objects are captured from multiple points of view. In object-wise experiments, images are divided according to the object categories. Objects in the test set have never been seen during training. This is to test the generalization ability of our method when it faces a new kind of object.

The evaluation of grasp detection accuracy and efficiency are summarized in Table 1. Our method achieves 98.9 and 98.3% accuracy on image-wise and object-wise detection tasks, respectively. Compared with the state-of-the-art RGB-D-based method (Yu et al., 2022b), our SOGD shows an improvement of +0.7 and +1.2% in

accuracy. The efficiency of our method is relatively low than SE-ResUNet (Yu et al., 2022b). In SE-ResUNet, a squeeze-and-excitation residual network is designed to predict the width and orientation of the grasp rectangle and the quality of the grasp outputs. It does not involve the detection of the target object. As a result, the complexity of SE-ResUNet is smaller than ours and their detection ability is not strong as ours. Similar results are observed when our SOGD is compared with Kumra et al. (2020). Compared with the state-of-the-art RGB-based method (Yu et al., 2022a) and RG-D-based method (Park et al., 2020), our SOGD shows a similar performance in the image-wise detection task, but a superior performance in the object-wise detection task. This is mainly because our SOGD not only learns possible grasp configurations but also the correspondences between the target object and grasp candidates. In this way, it has the ability to figure out what is the best grasp configuration for a specific kind of object. Therefore, when facing new objects, it can benefit from learned knowledge of the relationship between grasp configurations and objects.

TABLE 1 Grasp detection results on the Cornell Grasp Dataset.

Methods	IW/%	OW/%	FPS	Input
Chu et al. (2018)	94.4	95.5	8.3	RGB
Wang et al. (2021)	96.1	95.5	-	RGB
Asif et al. (2019)	96.7	-	-	RGB
Yu et al. (2022a)	**98.9**	**97.8**	**50.0**	RGB
Dong et al. (2021)	96.4	96.5	9.4	RG-D
Song et al. (2020)	95.6	97.1		RG-D
Zhang et al. (2019)	92.3	91.7	25.2	RG-D
Park et al. (2020)	**98.6**	**97.8**	**62.5**	RG-D
Jiang et al. (2011)	60.5	58.3	0.2	RGB-D
Lenz et al. (2013)	73.9	75.6	0.7	RGB-D
Chu et al. (2018)	96.0	96.1	8.3	RGB-D
Kumra et al. (2020)	97.7	96.6	**50.0**	RGB-D
Yu et al. (2022b)	98.2	97.1	40.0	RGB-D
SOGD (ours)	**98.9**	**98.3**	9.6	RGB-D

The best performance of methods within a same kind of input is shown as bold.

Visualization of typical grasp detection results is shown in Figure 5. The three lines in the figure are ground truth annotations, predicted grasp configurations of our SOGD, and the corresponding grasp quality predicted by our SOGD. For each detected object, our model outputs the best grasp rectangle for it. In this experiment, the quality is not the direct output from the grasp detection branch in our SOGD model. Our grasp detection branch outputs a score map for each grasp candidate that can be treated as the quality of the grasp, as mentioned in a previous study (Yu et al., 2022a,b). But our model includes an alignment module to learn the correspondences between predicted objects and grasp configurations. The quality is the product of the score map and the correspondences. It represents the success rate of a grasp if there is a detected target object. From the figure, it can be seen that our SOGD has a good ability in detecting grasp rectangles and the output quality map is able to provide a clear reason for the grasp decision.

4.2. Grasp detection results on Jacquard Dataset

The Jacquard Dataset (Depierre et al., 2018) is collected from a simulator with CAD models of the ShapeNet dataset. There are 54k images with 11k different kinds of objects in the dataset. A large number of samples facilitate our model training. However, we still use some data augmentation strategies (like random rotation) to increase

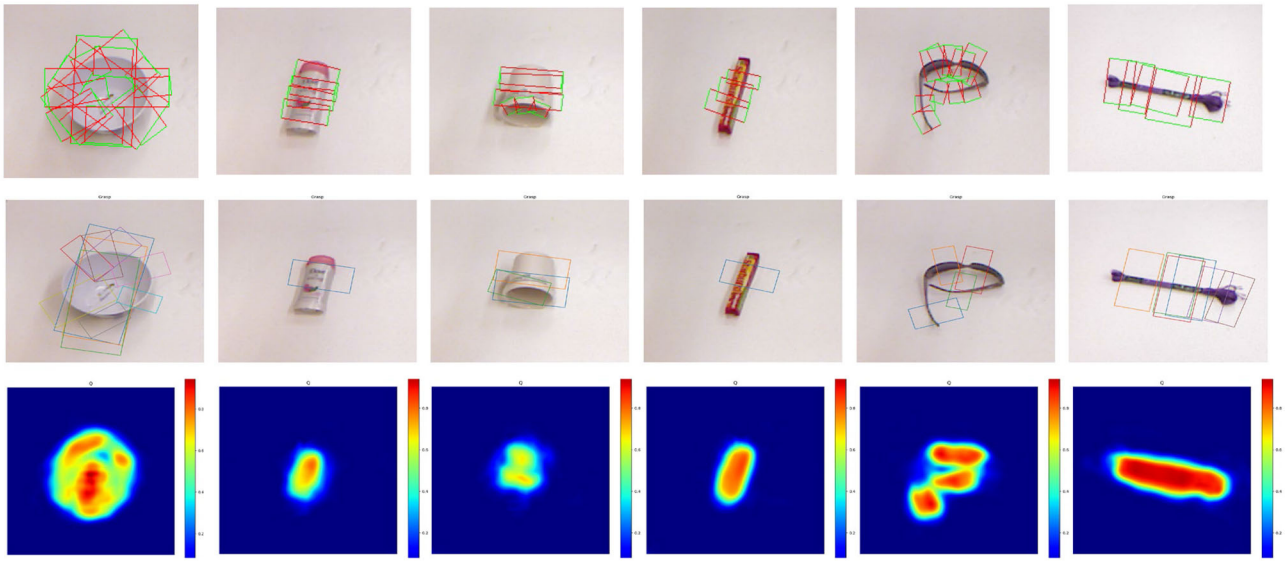

FIGURE 5
Typical grasp detection results of our SOGD on the Cornell Grasp Dataset. **Top** line is ground truth annotation; **middle** line is the prediction of our SOGD; and **bottom** line is the estimated grasp quality of the corresponding detection.

the robustness of the learned model. The resolution of images in this dataset is 1,024 × 1,024. We down-sample the original image to a size of 512 × 512 for both training and testing. Unlike results on the Cornell Dataset, we report the overall accuracy of the grasp detection results on the Jacquard Dataset.

Table 2 reports the performance of our SOGD against the state-of-the-art methods. Our SOGD shows a 99% accuracy on the Jacquard Dataset, which is higher than both RGB-based and RG-D-based state-of-the-art methods. Compared with MASK-GD (Dong et al., 2021), our method achieves a +1.4% performance boost. MASK-GD also involves pre-processing for background removal. Background removal is treated as an image segmentation problem and a deep network is trained for it in MASK-GD. This strategy has the potential to recognize the foreground targets from the cluttered scenes, and may also suffer the problem of limited generalization ability. In addition, our model has an additional alignment module to learn the relationship between object candidates and grasp candidates, while MASK-GD cannot. Compared with other grasp detection methods, the improvement of our SOGD is more significant.

Figure 6 shows some typical grasp detection results of our SOGD on this dataset. Both the detected grasp configurations and the quality maps are visualized to give a better understanding of the results. From the figure, it can be seen that our SOGD can well detect grasp candidates and outputs a reasonable quality map on this dataset.

TABLE 2 Grasp detection results on the Jacquard Dataset.

Methods	Accuracy/%	Input
Zhou et al. (2018)	91.8	RGB
Zhang et al. (2019)	90.4	RGB
Dong et al. (2021)	**97.1**	RGB
Zhou et al. (2018)	92.8	RG-D
Zhang et al. (2019)	93.6	RG-D
Dong et al. (2021)	**97.6**	RG-D
SOGD (ours)	**99.0**	RGB-D

The best performance of methods within a same kind of input is shown as bold.

4.3. Object detection results

For object detection evaluation, our SOGD model is trained and tested on the two datasets mentioned earlier. We use the Labelme tools from MIT to manually annotate bounding boxes and class labels on the Cornell Dataset. To prevent overfitting, the pre-trained

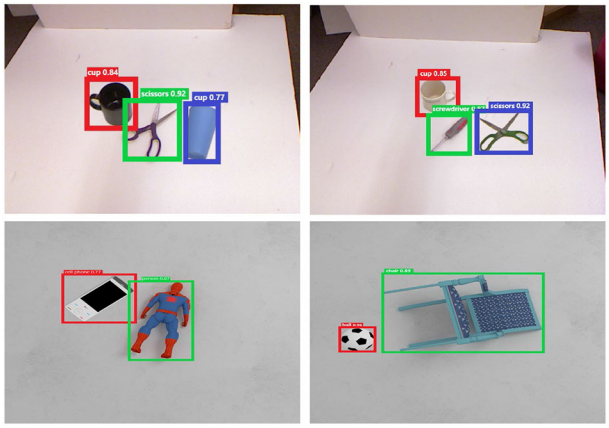

FIGURE 7
Typical object detection results.

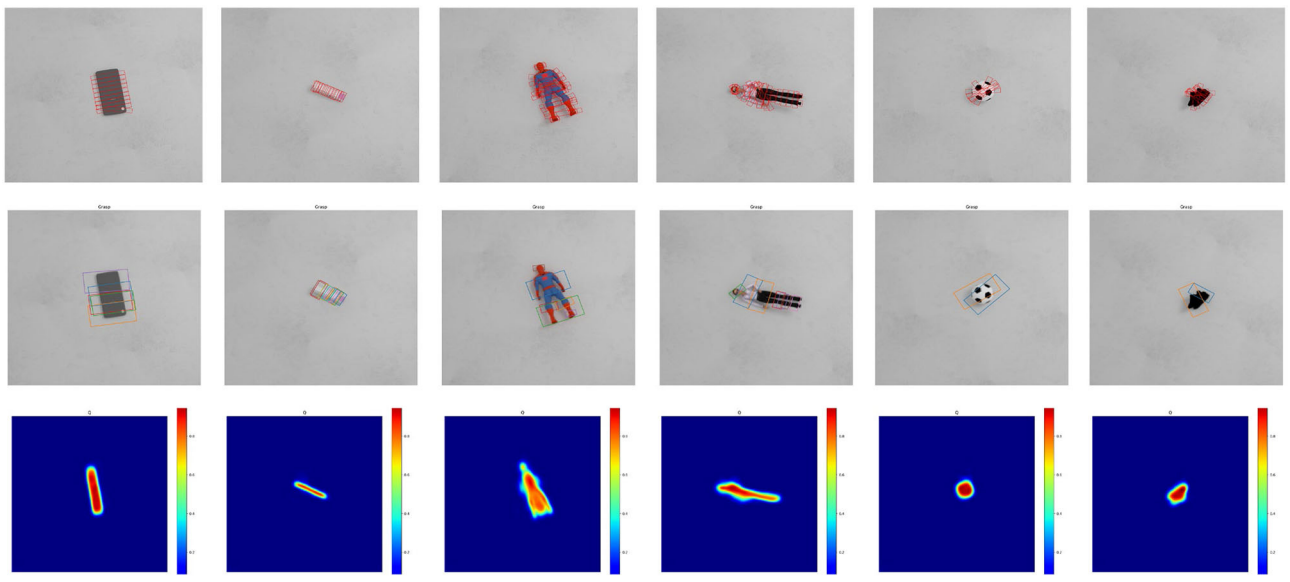

FIGURE 6
Grasp detection results of our SOGD on the Jacquard Dataset. **Top** line is ground truth annotation; **middle** line is the prediction of our SOGD; and **bottom** line is the estimated grasp quality of the corresponding detection.

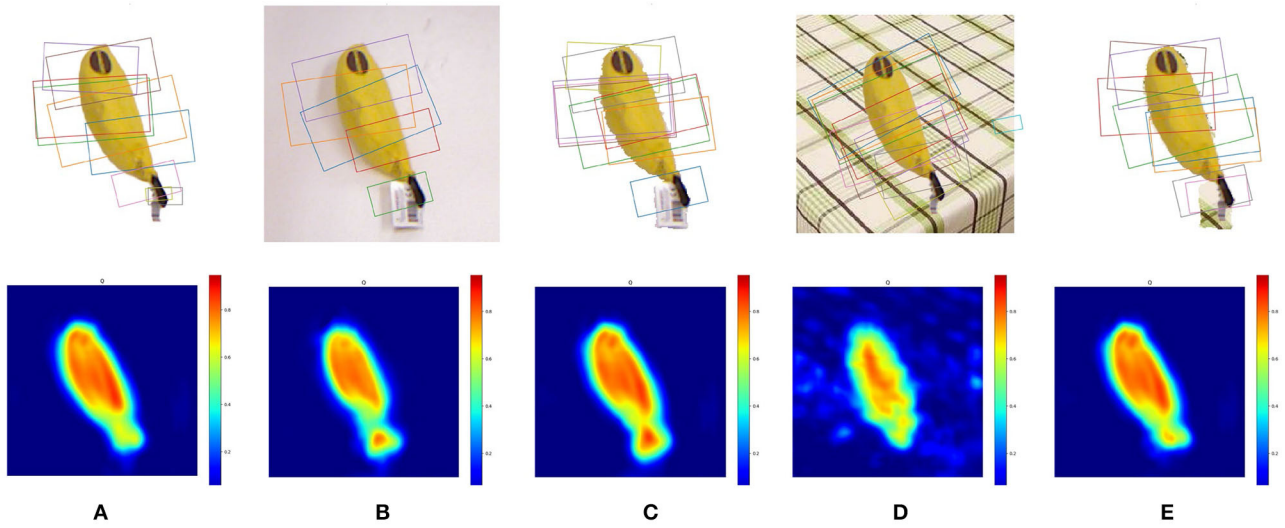

FIGURE 8
Grasp detection results on images with various kinds of backgrounds. **(A)** SOGD without background removal (BR) on images without background. **(B)** SOGD without BR on original images from Jacquard Dataset. **(C)** SOGD with BR on the original images. **(D)** SOGD without BR on images with the background being replaced. **(E)** SOGD with BR on images with the background being replaced. From left to right the background of the image becomes more cluttered.

parameters of ResNet-50 are fixed for the backbone and several data augmentation strategies are involved, such as rotation, translation, flip, random crop, and illumination change. Typical experimental results are shown in Figure 7. From the figure, it can be seen that after removing the background boundaries foreground objects are much easier to detect, releasing the burden of the object detection branch and resulting in more accurate bounding box predictions. In our experiments, though we pay more attention to the prediction accuracy of where the object is, the classification confidence of our SOGD is almost above 0.75 on the two datasets. The above-mentioned observations show that our SOGD model has a good performance in detecting objects from a cluttered scene, especially in identifying the boundaries of the objects.

4.4. Discussion on background removal

From the results in Tables 1, 2, it has already been seen that our SOGD has superior performance than existing methods without background removal (Zhang et al., 2019; Kumra et al., 2020; Yu et al., 2022b). But to investigate how much the background removal contributes to this performance boost, we conduct an ablation study on this pre-processing. The Jacquard Dataset is used in this experiment since it provides a ground truth mask for foreground target objects. With this mask, we generate two additional types of images from the original dataset to test the performance of our SOGD on it. The first one is to filter out backgrounds with the ground truth mask. The second one is to fill the background with a cluttered background. To achieve this, we download a number of images from the Internet as the background image datasets and randomly choose one to replace the background images from the Jacquard Dataset.

We provide a comparison among five types of grasp detection configurations: (1) SOGD without background removal on images with background filtered out, (2) SOGD without background removal on the original image, (3) SOGD with background removal on the original image, (4) SOGD without background removal on images with background being replaced, and (5) SOGD with background removal on images with background being replaced. Typical results are shown in Figure 8. The background of the scene in Figure 8 becomes more cluttered from left to right. From the figure, we observe that the predicted grasp configurations will be much better when the background is removed from the image.

5. Conclusion and future study

This study is focused on the problem of grasp detection from an RGB-D image. Unlike previous methods, we solve this problem by simultaneously detecting the target object and the corresponding grasp configurations. This is motivated by the fact that when grabbing an object, we humans first identify where the object is and then make a decision on which part of the object to grab. To this end, a novel neural network SOGD together with its learning method is proposed. In SOGD, object and grasp configurations are first detected by two separate branches, and then the relationship between object candidates and grasp configurations is learned by an alignment module. The best grasp configuration is predicted according to the grasp score and its correspondence to the target object. Our method is tested on two publicly available datasets. A series of experiments are conducted and both qualitative and quantitative experimental results are presented. The results demonstrate the validity and practicability of our method.

To deal with grabbing in a cluttered scene, a pre-processing for background removal is designed. Unlike previous methods where background removal is treated as an image segmentation

problem, we propose to leverage the prior knowledge that objects to be grabbed are often placed on a 3D plane. Therefore, we adopt a RANSAC-based plane fitting method to detect the largest 3D plane in the scene. All pixels laid in or under the plane are considered background. The experimental results show that our strategy makes grasp detection more robust in cluttered environments.

The stacked scene is not considered in this research. In daily life cases, it is common that objects to be grabbed are laid on top of each other. This is more challenging for the grasp detection method because it has to figure out the execution order of the predicted grasp configurations. This is an interesting topic for our future study. In addition, the kind of object for model training is limited. It has to face a large number of unknown objects when the learned model is deployed to real devices. It is interesting to extend our model with the life-long learning ability after deployment. We will explain it in our future study.

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.

Author contributions

YuaL contributed the main ideas and designed the algorithm. YuhL organized the database. LX performed the statistical analysis. YZ wrote the first draft of the manuscript. YZ, LX, YuhL, and YuaL wrote sections of the manuscript. All authors contributed to manuscript revision, read, and approved the submitted version.

Conflict of interest

YZ, LH, and YuaL were employed by China Tobacco Sichuan Industrial Co., Ltd. YuhL was employed by Qinhuangdao Tobacco Machinery Co., Ltd.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Ainetter, S., and Fraundorfer, F. (2021). "End-to-end trainable deep neural network for robotic grasp detection and semantic segmentation from RGB," in *Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA)* (Sanya, China: IEEE), 13452–13458.

Asif, U., Tang, J., and Harrer, S. (2019). "Densely Supervised Grasp Detector (DSGD)," in *Proceedings of the AAAI Conference on Artificial Intelligence* (Hawaii, USA: AAAI), 33, 8085–8093.

Cheon, J., Baek, S., and Paik, S. B. (2022). Invariance of object detection in untrained deep neural networks. *Front. Comput. Neurosci.* 16, 1030707. doi: 10.3389/fncom.2022.1030707

Chhabra, M., Ravulakollu, K. K., Kumar, M., Sharma, A., and Nayyar, A. (2022). Improving automated latent fingerprint detection and segmentation using deep convolutional neural network. *Neural Comput. Appl.* 2022, 1–27. doi: 10.1007/s00521-022-07894-y

Chu, F. J., Xu, R., and Vela, P. A. (2018). Real-world multiobject, multigrasp detection. *IEEE Robot. Autom. Lett.* 3, 3355–3362. doi: 10.1109/LRA.2018.2852777

Depierre, A., Dellandréa, E., and Chen, L. (2018). "Jacquard: A large scale dataset for robotic grasp detection," In Proceedings of the *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Madrid, ES: IEEE), 3511–3516.

Dong, M., Wei, S., Yu, X., and Yin, J. (2021). Mask-GD Segmentation Based Robotic Grasp Detection. *Comp. Commun.* 178, 124–130. doi: 10.1016/j.comcom.2021.07.012

Ge, Z., Liu, S., and Wang, F. (2021). Yolox: exceeding yolo series in 2021. *arXiv preprint arXiv:2107.08430*

Georgakis, G., Karanam, S., Wu, Z., and Kosecka, J. (2019). "Learning Local RGB-to-CAD Correspondences for Object Pose Estimation," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)* (Seoul, Korea (South): IEEE), 8966–8975.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). "Deep residual learning for image recognition," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (Las Vegas, USA: IEEE), 1063–6919. doi: 10.1109/CVPR.2016.90

Huang, Z. J., Hui, B. W., and Sun, S. J. (2022). An infrared sequence image generating method for target detection and tracking. *Front. Comput. Neurosci.* 16, 930827. doi: 10.3389/fncom.2022.930827

Jiang, Y., Moseson, S., and Saxena, A. (2011). "Efficient grasping from rgbd images: Learning using a new rectangle representation," in *Proceedings of the 2011 IEEE International Conference on Robotics and Automation (ICRA)* (Shanghai, CHN: IEEE), 3304–3311.

Jiang, Z., Zhu, Y., Svetlik, M., Fang, K., and Zhu, Y. (2021). "Synergies between affordance and geometry: 6-DoF grasp detection via implicit representations," in *Robotics: Science and Systems XVII* (Robotics: Science and Systems Foundation).

Khan, A., Khan, A., Ullah, M., Alam, M. M., Bangash, J. I., Suud, M. M., et al. (2022). A computational classification method of breast cancer images using the VGGNet model. *Front. Comput. Neurosci.* 16, 1001803. doi: 10.3389/fncom.2022.1001803

Kumra, S., Joshi, S., and Sahin, F. (2020). "Antipodal Robotic Grasping using Generative Residual Convolutional Neural Network," In *Proceedings of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Prague, CZ: IEEE), 9626–9633.

Lenz, I., Lee, H., and Saxena, A. (2013). *Deep Learning for Detecting Robotic Grasps*. London, England: SAGE Publications Sage UK.

Liang, H., Ma, X., Shuang, L., Grner, M., and Zhang, J. (2019). "PointNetGPD: detecting grasp configurations from point sets," In *Proceedings of the 2019 International Conference on Robotics and Automation (ICRA)* (Montreal, CAN: IEEE), 3629–3635.

Lin, T. Y., Goyal, P., Girshick, R., He, K., and Dollár, P. (2017). Focal loss for dense object detection. *IEEE Trans. Patt. Anal. Mach. Intell.* 42, 318–327. doi: 10.1109/TPAMI.2018.2858826

Motwani, A., Shukla, P. K., Pawar, M., Kumar, M., Ghosh, U., Al Numay, W., et al. (2022). Enhanced framework for COVID-19 prediction with computed tomography scan images using dense convolutional neural network and novel loss function. *Comput. Electr. Eng.* 105, 108479. doi: 10.1016/j.compeleceng.2022.108479

Park, D., Seo, Y., Shin, D., Choi, J., and Chun, S. Y. (2020). "A single multi-task deep neural network with post-processing for object detection with reasoning and robotic grasp detection," in *Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA)* (Paris, FR: IEEE), 7300–7306.

Pas, A. T., Gualtieri, M., Saenko, K., and Platt, R. (2017). Grasp pose detection in point clouds. *Int. J. Robot. Res.* 36, 1455–1473. doi: 10.1177/0278364917735594

Redmon, J., and Farhadi, A. (2018). Yolov3: an incremental improvement. *arXiv preprint arXiv:1804.02767*

Shailendra, R., Jayapalan, A., Velayutham, S., Baladhandapani, A., Srivastava, A., Kumar Gupta, S., et al. (2022). An IoT and machine learning based intelligent system for the classification of therapeutic plants. *Neural Process. Lett.* 2022, 1–29. doi: 10.1007/s11063-022-10818-5

Singh, M., Shrimali, V., and Kumar, M. (2022). Detection and classification of brain tumor using hybrid feature extraction technique. *Multimedia Tools Appl.* 2022, 1–25. doi: 10.1007/s11042-022-14088-0

Song, Y., Gao, L., Li, X., and Shen, W. (2020). A novel robotic grasp detection method based on region proposal networks. *Robot. Cim-Int. Manuf.* 65, 101963. doi: 10.1016/j.rcim.2020.101963

Sundermeyer, M., Mousavian, A., Triebel, R., and Fox, D. (2021). "Contact-GraspNet: Efficient 6-DoF Grasp Generation in Cluttered Scenes," in *2021 IEEE International Conference on Robotics and Automation (ICRA)* (Xi'an, China: IEEE), 13438–13444.

Wang, D., Liu, C., Chang, F., Li, N., and Li, G. (2022). High-performance pixel-level grasp detection based on adaptive grasping and grasp-aware network. *IEEE T. Ind. Electron.* 69, 11611–11621. doi: 10.1109/TIE.2021.3120474

Wang, S., Jiang, X., Zhao, J., Wang, X., Zhou, W., Liu, Y., et al. (2019). "Efficient fully convolution neural network for generating pixel wise robotic grasps with high resolution images," in *Proceedings of the 2019 IEEE International Conference on Robotics and Biomimetics (ROBIO)* (Yunnan, CHN: IEEE), 474–480.

Wang, Y., Zheng, Y., Gao, B., and Huang, D. (2021). "Double-Dot Network for Antipodal Grasp Detection," in *Proceedings of the 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Prague, CZ: IEEE), 4654–4661.

Wood, J. A. (2009). The Darknet: A digital copyright revolution. *Rich. JL Tech.* 16, 1. http://jolt.richmond.edu/v16i4/article14.pdf

Yang, D., Tosun, T., Eisner, B., Isler, V., and Lee, D. (2021). "Robotic Grasping through Combined Image-Based Grasp Proposal and 3D Reconstruction," in *2021 IEEE International Conference on Robotics and Automation (ICRA)* (Xi'an, China: IEEE), 6350–6356.

Yu, S., Zhai, D. H., and Xia, Y. (2022a). EGNet: Efficient Robotic Grasp Detection Network. *IEEE T. Ind. Electron.* 2022, 1–1. doi: 10.1109/TMECH.2022.3209488

Yu, S., Zhai, D. H., Xia, Y., Wu, H., and Liao, J. (2022b). SE-ResUNet: A novel robotic grasp detection method. *IEEE Robot. Autom. Lett.* 7, 5238–5245. doi: 10.1109/LRA.2022.3145064

Zhang, H., Lan, X., Bai, S., Zhou, X., Tian, Z., Zheng, N., et al. (2019). "ROI-based Robotic Grasp Detection for Object Overlapping Scenes," in *Proceedings of the 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Macau, CHN: IEEE), 4768–4775.

Zhang, H. B., Lan, X. G., Zhou, X. W., Tian, Z. Q., Zhang, Y., Zheng, N. N., et al. (2018). Robotic grasping in multi-object stacking scenes based on visual reasoning. *Scientia Sinica Technologica*. 48, 1341–1356. doi: 10.1360/N092018-00169

Zhou, X., Lan, X., Zhang, H., Tian, Z., Zhang, Y., Zheng, N., et al. (2018). "Fully convolutional grasp detection network with oriented anchor box," In *Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (Madrid, ES: IEEE), 7223–7230.