6-DoF occluded object semantic grasp planning with de-occlusion instance segmentation

Zhongming Huang*
School of Electronic and Information Engineering
Tiangong University
Tianjin, China
reavenhuang@ieee.org

Shangyun Yang
School of Electronic and Information Engineering
Tiangong University
Tianjin, China
yangshangyun@tiangong.edu.cn

Abstract: Thanks to the previous researches, computer vision has now endowed 6-DoF robot arms with the intelligence to plan its best trajectory to a given target. One of the problems that the 6-DoF semantic grasp planning still faces is the solution to grasping occluded targets in a cluttered scene. In such scenario, the robot arm may not be available to grasp the target in one shot. The current potential attempts to grasp an occluded object includes re-arranging the cluttered scene in the hope to expose the target object for a more viable trajectory. However, such methods require multiple viewpoints to model the scene for making global re-arranging plans. Also, the process of re-arranging would take considerable steps to achieve. In our work, we use our decision-making algorithm to combine occlusion prediction (BCNet) with grasp pose planning algorithm (GraspNet), enabling the robot arm to understand the relative positions of each object in a scene. Our method is based on a single view point, so that we do not require the robot arm to look all around the scene. In addition, our method is target-motivated, which means we only grasp relevant objects instead of re-arranging all objects in the scene, providing a novel efficient solution to grasp occluded targets.

Keywords: 6-DoF Semantic Grasp Planning, Instance Segmentation, Occlusion Prediction, Robotic Grasping, Vision and Perception.

I. INTRODUCTION

Semantic grasping planning for 6-DoF robot arms has been a rising topic of robotic intelligence, this is a process involving RGB-D depth image processing, object localization, grasp pose generation and trajectory planning. Current works [1] have achieved robust algorithms to solve the problem that how the robot could understand the layout of a certain scenario, as well as how to approach the target object in a correct movement and position. So, that overcomes the barriers of the mentioned processes, everything seemed to be a no-problem. But is this pleasure a real relief to researchers? The answer is negative, after the applications of semantic grasping expanded beyond the laboratory environment, reaching to real-world occasions. One difference between pure lab environments and practical situations is the layout of the scene where the robot seeks for a target to conduct grasps. In real world scenes, objects are often arranged tightly to each other, consisting a cluttered layout which is difficult to conduct traditional semantic grasping [2] such scenes are shown in Fig. 1 . There are several obstacles contributing to the difficulty for robots to conduct a successful semantic grasp within such cluttered scenes: a) the difficulty to spot the target object given the target has been occluded by cluttered obstacles; b) the unavailability for the robot arm to directly approach the target object successfully. To solve the problem, previous researchers have made great efforts such as re-arranging the scene to make the layout more sparse and convenient for a robot arm to move around [3]. However, such method requires great amount of robotic observation around the layout, which involves heavy load of computation since the robot has to reconstruct the whole scene before determining the movements of each object in the scene. In addition, this re-arranging process also requires a considerable number of grasp movements, in order to place each object to the expected location. Admittedly, such re-arranging methods bring advantages in other applications as robotic tidying, it is not an optimal choice to simplify semantic grasping in a cluttered scene.

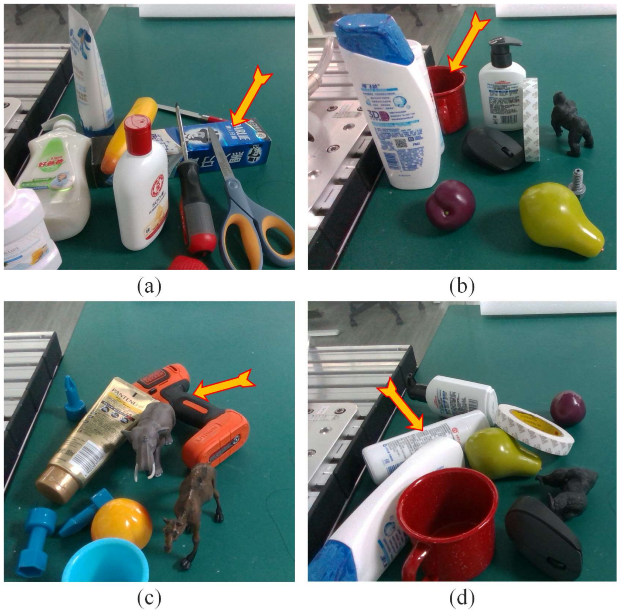

Fig.1 Real-world grasps scenes. Images are by courtesy of GraspNet-1Billion dataset. Occluded target objects in the above four scenes are indicated by yellow arrows: a) the toothpaste package; b) the red cup; c) the electric drill; d) the white bottle.

In our work, we are about to try another method to enable the robot arm to finally reach its target and make a successful

* Corresponding author

grasp. We focus on removing the objects which directly occludes our target, instead of trying to divide and re-arrange all objects in the scene. By the word removing, we mean grasping and placing the occluding objects out of the scene. I order to have our robot understand the spatial relationship of the scene without exhausting observance, we make use of RGB-D data from a single camera viewpoint to exploit the depth relationship while segmenting objects and predicting occlusion area. In the actual implementation of grasping, we adapt a grasp pose detection network [4] to provide potential viable grasp positions and gestures on each object in the scene. In order to select the best next grasp, which could be the occluding objects or occluded target, we design and test our decision-making algorithm. This decision-making algorithm makes use of both the spatial and occlusion relationships from the segmentation-and-occlusion-prediction network (BCNet) [5] and the grasp confidence of the grasp detection network, to generate scores of each object in the scene. The score given by our algorithm is an overall estimation of the viability of a grasp and the profit the grasp can bring after removing the object. Our algorithm always chooses to grasp the object with the highest score until it grasps the target object.

In our decision-making algorithm, we gather parameters from previous two networks and weigh them to conclude the best next grasp. This decision-making algorithm is always pursuing the optimal grasp on the occluding object and the target, considering factors including the pose confidence, the area of occlusion. When there are no direct reliable grasp poses on the target, our decision-making algorithm will try to manipulate the robot to grasp the best occluding object until the target is graspable at one shot. We adjusted our decision-making algorithm on the CoppeliaSim virtual grasping environment to finetune the parameters. After that, we tested our model on randomly generated cluttered scenes with the target object occluded and harvested over 4 times success rate improvement. To discuss our criteria to mark the object as the "best next grasp" as well as our standard for an efficient grasp, we will expand our designs in Part III.

II. Related Works

A. Visual Grasp Pose Detection

The most important approach for a robot arm to percept the surrounding environment is by looking around. It may sound easy, but it in fact has evolved a lot before we can now generate 6-DoF grasp poses [6] . The easiest method for a robot to see the scene with objects to be grasped is pure camera video streams. Since the video streams are consisted of consecutive RGB image frames, the robot is not able to feel the cubical appearance of objects. Hence, the grasp pose proposals generated from such perception are almost 2D poses. This means the robot can only look from above and vertically to the scene. Redmon and Angelova (2015) [7] trained a single-stage CNN to detect 2D planar grasps and conduct object classification at the same time. Such method is fast, and is suitable for robot arms which are not that flexible. However, observations from such planar images is not sufficient for robots to judge whether the edges of the proposed grasp poses are indeed reliable for the robot to grasp tight. Also, planar grasp pose proposals will limit the possibility for robots to approach target objects from a more viable trajectory, since the robot arm given a planar grasp pose can only grasp vertically from the above. Zhang et.al. (2017) proposed [8] a multi-modal fusion approach to provide 2D grasp poses with the combination of RGB image and depth data. The aid of depth information is critical to understanding the accurate margins and the shape of edges before proposing poses, and this expanded its application from lab environment where pure background are guaranteed to many real-world polychrome environments. Moreover, depth data also acts as a new criterion when detecting grasps on strange or unknown objects.

As robots with higher flexibility gradually prevails, mainstream researches focus on how to give poses in a more natural and un-limited manner. Such pose detection methods generate 6-DoF poses from RGB-D data and allows robot arms to approach the targets in various trajectories. Especially in its application in dense clutters, robots can benefit from the variety of approach angles since 2D grasps may not be practical. Gualtieri et.al. (2016) [9] proposed a 6-DoF grasp pose detection algorithm which is capable of scene with cluttered object, the robot in their work achieved over 90% success rate when working in active mode to eliminate the clutters. We choose to adapt from GraspNet [4], which also generates 6-DoF grasp poses, but in a more general scale. Since GraspNet was pre-trained and tuned on various objects and over cluttered scene, it has shown advanced generality to real-world scenarios, providing wider range of objects that could be grasped.

B. Occlusion Prediction Methods

Another factors that inspired robotic grasping in occluded scenarios are the advancements of occlusion prediction algorithms. Occlusion prediction, also known as de-occlusion methods, is now more of a branch under instance segmentation algorithms. Before the prosperity of instance segmentation algorithms, Chen, X., & Yuille, A. L. (2015) [10] tried to de-occlude human arms in collective photos based on the specific features of a human torso, which is not dependent on the segmentation of different people. Such de-occlusion methods are based on the abstract modeling of specific structures and is not general enough to be transferred to common object de-occlusions. As to the aspect of occlusion prediction with instance segmentation, Chen, Y.-T., Liu, X., & Yang, M.-H. (2015) [11] conducted an early try on a scene of multiple objects. The occlusion area predictions are based on the margins of segmentation, and the network will reason the overlapping area pixel-wise after learning human-labelled annotations which are also masks in pixel level. By predicting the occlusion areas, the original object being occluded in the scene can be reconstructed. Zhan et.al. (2020) [12] also pointed out the possibility for comprehending the spatial relationship through de-occlusion and reconstruct each segmented object in the scene. We choose BCNet proposed by Ke et.al. in 2021 [5], it is a bi-layer occlusion prediction network, where two parallel networks are predicting the original margins of the two objects in the front scene and the background. Its leading performance in correctly predicting the margins of small overlapping areas brings benefit in our grasp scenes where small objects are dominant.

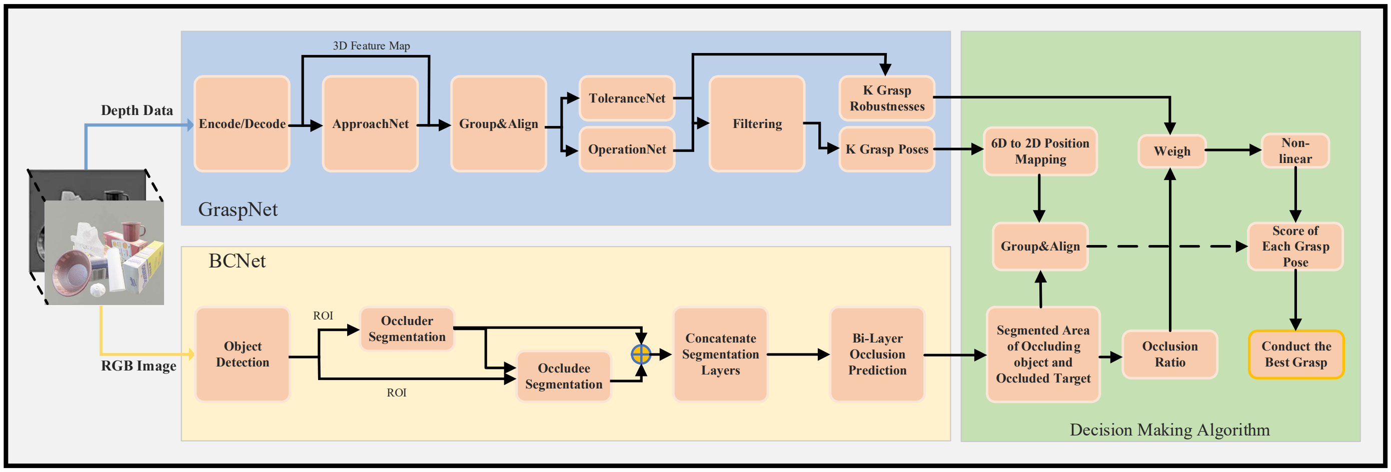

Fig. 2 Overview of Our Algorithm. The input is an RGB-D image collected from the fixed camera in the simulated scene. The GraspNet and BCNet will process the depth channel and the RGB color channel respectively. The grasp implementation is determined by our decision-making algorithm.

III. Our Method

A. Overview

The general workflow is shown in Fig. 2, we use BCNet and GraspNet as our parallel front ends, and we extract some of the variables from the two networks as our criteria for deciding the next best grasp.

B. Grasp Pose Detection

To endow our model the ability detect grasp poses, we adapt GraspNet [4] , a novel and precise algorithm. And we utilize its benchmark GraspNet-1Billion as our dataset where we adjust parameters and test our algorithm. The GraspNet uses PointNet++ [13] as its backbone network and is capable of processing RGB-D data captured from the object scene and exploit information from both the RGB image channels and the depth channel. It was pre-trained and evaluated from massive 3D object models and has shown preferable performance in pose richness and speed when tested on the cluttered object scene. To feature on actual test result, this network brings up to 29.88 average precision which is leading among other algorithms being compared [4]. As shown in Fig.2, the point cloud depth image is first encoded and decoded into a 3D feature map, and then be passed to ApproachNet to detect possible grasp approach angles and points (approach vectors described in [4]). After that, the approach vectors are grouped with the correlating feature points from the previous 3D feature map, at this step, each approach vector points at a certain feature point. In the next step, the approach vectors are evaluated parallelly by OperationNet and ToleranceNet, which respectively predicts the grasp pose parameters and their grasp robustness. Finally, it outputs filtered grasps poses.

In our work, we do not need the robot to directly conduct grasps based on those poses. We here collect some important parameters and the final grasp decision will be based on our choice of the next best grasp (NBG) in section D. We first run it to detect every possible grasp in our scene, assume there are G_{global} viable grasp poses after filtering, upon all the objects detected. And we make use of the output grasp robustness $R, sized\ 1 \times G_{global}$ which are aligned with the viable G_{global} grasp poses. We temporarily save the corelating grasp pose parameters $P_{global}, sized\ 6 \times G_{global}$, for further selection of NBG.

C. Occlusion Prediction

To understand the occlusion relationship, we need another network which is capable of instance segmenting and occlusion prediction. BCNet [5], an efficient and reliable new method to help with understanding the exact margins of each object and estimates the overlapping area of different objects in an image, comes to our sight. It intakes RGB image data and outputs segmentations of each object as well as the predicted margins of occluded objects. We will combine depth data and its output margins to help the robot understand which objects are occluding the target and how much we can expose target's occluded area by removing one of the occluding objects. The area that could be exposed by removing occluding objects is helpful in indicating the necessity of conducting a removing grasp in our decision-making algorithm.

When the RGB image is input to BCNet, this network first detects the region of interest (ROI), and further procedures are based on the clipped ROI out of the whole image. The ROI is processed to predict the occluding objects (occluder) first, and the shape and margin of the occluder will help predict the occluded objects (occludee). Finally, the two layers of occluder and occludee will be added together. We need the pixel-wise segmentation of each object, and more than that, we need the overlapping pixel number of different object segmentations. Let's assume the area of ROI to be $\mathrm{A}_{\mathrm{global}}, W \times H$, where W and H are equal to 28, as the width and height of the ROI in pixels. And we assume $\mathrm{A}_{\mathrm{masks}}, \mathrm{N}_{\mathrm{obj}} \times W \times H$, where $\mathrm{N}_{\mathrm{obj}}$ is the number of segmented objects and the element of $\mathrm{A}_{\mathrm{masks}}$ is 0 or 1, we can

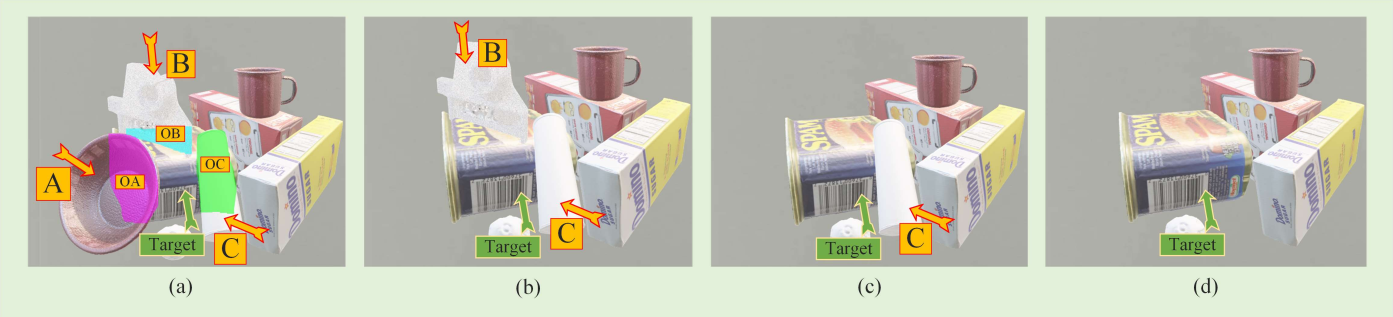

Fig.3 The Process of De-occlusion. As shown in sub-figure (a), A, B and C are the three occluding objects (occludees) indicated by yellow arrows and marks; the target object is the blue can indicated by a green arrow and mark. The occlusion areas of A, B and C are predicted and indicated by OA, OB and OC which are illustrated in purple, cyan and green respectively. Our decision-making algorithm will determine to grasp object A first.

find the segmentation of the $\mathrm{i^{th}}$ object by looking for element 1 in $\mathrm{A_{mask}(i)}$. We first obtain the pixel-wise area $\mathrm{A_{object}}$, $1 \times N_{obj}$ of each segmented object:

$$\mathrm{A_{object}(i)} = \sum_{\mathrm{m=1,n=1}}^{\mathrm{W,H}} \mathrm{A_{masks}(i,m,n)} \qquad (1)$$

We then derive a occlusion area matrix $\mathrm{A_o}, \mathrm{N_{obj}} \times \mathrm{N_{obj}}$ from the equation below:

$$\mathrm{A_o}(i,j) = \sum_{\mathrm{m=1,n=1}}^{\mathrm{W,H}} \left(A_{masks}(i,m,n) + A_{masks}(j,m,n) \ - 1\right) \ (2)$$

In Eq. (2), $\mathrm{A_o(i,j)}$ represents the overlap pixels of object i and j , and it is a symmetrical matrix.

D. Decision-making Algorithm

Since we have obtained the occlusion area matrix $\mathrm{A_o}$, the pixel-wise area of each object $\mathrm{A_{object}}$, we can further determine the occlusion ratio $O, \mathrm{N_{obj}} \times \mathrm{N_{obj}}$:

$$\mathrm{O(i,j)} = \frac{\mathrm{A_o(i,j)}}{\mathrm{A_{object}(j)}} \qquad (3)$$

In Eq.3, $O\mathrm{(i,j)}$ describes the ratio of the occlusion area resulted from the $\mathrm{i^{th}}$ and the $\mathrm{j^{th}}$ object to the area of the $\mathrm{j^{th}}$ object. And we always have $O\mathrm{(i,j)} \leq 1$. O is an important indicator about the exposed area it can bring if we grasp certain object, leaving the behind object un-occluded.

We then determine the target object and those objects occluding. If our target object is the $\mathrm{m^{th}}$, we can find all the objects occluding it by calling:

$$\mathrm{O(i,m)} \geq \partial \qquad (4)$$

In Eq.4, ∂ is a lower limit that we will finetune in Part IV. Each i that satisfies Eq.4 will indicate an occludee indexed $\mathrm{i^{th}}$. We call the object occluding our target the Related Object (RO).

To know which object that the grasp poses in section A belong to, we have to map the 6D grasp poses to planar (2D) space of the RGB image. Assume $t \in \mathrm{R^3}$ represents the grasp pose spatial location $t\mathrm{(x,y,z)}$, and $\mathrm{t'} \in \mathrm{R^2}$ to indicate the corresponding planar position $\mathrm{t'}(w,h)$ from the RGB image viewpoint, we have the following transformation formula:

$$w = (\mathrm{t - C}) \cdot (-\sin(\theta), \cos(\theta), 0);$$

$$\mathrm{h} = (\mathrm{t - C}) \cdot \left(\cos(\theta) \cdot \sin(\phi), \sin(\theta) \cdot \sin(\phi), \cos(\phi)\right) \ (5)$$

In Eq.5, we have following quantities:

$$C = \left(\cos(\theta) \cdot \cos(\phi), \sin(\theta) \cdot \cos(\phi), \sin(\phi)\right) \cdot r;$$

$$r = |t| = \sqrt{\mathrm{x^2 + y^2 + z^2}} \qquad (6)$$

Note that angles θ, ϕ are known from the camera viewpoint angle. Now, we have obtained the grasp pose location $\mathrm{t'}(w,h)$ in the image plane. The grasps of a certain object can now be called through searching $\mathrm{A_{masks}}$, and $\mathrm{A_{masks}(i,w,h)} = 1$ means the grasp pose is belonged to the $\mathrm{i^{th}}$ object.

Once we have obtained RO's ratio of the occlusion area, we can establish our scoring formula which takes consideration on the grasp robustness R and ratio of occlusion O, and we use S, $1 \times \mathrm{N_{obj}}$ to represent the output score:

$$\mathrm{S(i)} = \frac{1}{1 + \mathrm{e}^{-[O\mathrm{(i,m)} \times 5 - 2.5]}} \cdot \mathrm{R(i)} \qquad (7)$$

In Eq.7, the target object is indexed m, and our next best grasp (NBG) is defined by the object with the highest score. The de-occlusion grasp process is illustrated in Fig.3.

IV. Experiment

A. Dataset Preparation

In order to train the BCNet to be more adapted to real-world grasp scene, we synthesized 500 occlusion scene images for its transfer learning. Our synthesized images are based on combinations of basic object models from GraspNet-1Billion dataset [4]. Each synthesized image has real and complete object contours of occluded and partially occluded objects, allowing explicit modeling of occluded relationships between occluded regions and occluded objects. We use this synthesized dataset to complete the training of BCNet, so as to better judge the effect of overlapping areas and make our network prediction more accurate.

For parameter tuning and model testing, we use a grasp simulator to generate random scenes based on object models in [4]. These objects are suitable for the size of the grippers, and in terms of shape, texture, size, material, and diversity. And this object dataset can better simulate the real-world grasps scenes. We use this objects in this dataset to create a more intensively occluded scene by generating dense clutters. In each scene, we randomly put in 6 to 10 everyday objects. Table.1 shows the basic information of our dataset.

TABLE 1 DATASET INFORMATION

Synthesized Occlusion Images	500
Occlusions / Image	~4
Objects / Grasp Scene	6~10
Scenes / Test Epoch	100

B. Environment Setup

Our grasp experiment is based on CoppeliaSim V4.2.0 rev5, where a virtual grasp environment is provided, with the BCNet framework based on PyTorch 1.4.0. And we select UR5 robot arm as our grasp actuator. The computation is based on a RTX2080 GPU and Intel i7 CPU. In the experiment, the camera viewpoint is fixed from angled where the sight is inclined downward, the robot will execute grasps based on the perception from the camera.

C. Parameter Finetuning

We first finetune BCNet, which achieved 31.75 AP after the network was sufficiently converged. Compared with other image segmentation networks, our finetuned BCNet showed stronger sensitivity in predicting overlapping regions. We expect that it can be used to accurately judge the area between occlusions and target objects, so as to lay a foundation for the subsequent adjustment of model parameters.

We adapted the GraspNet framework and its pre-trained parameters, and we finetune the parameter ∂ in our decision-making algorithm based on the pre-trained GraspNet. We test different values of ∂ in the simulated grasp scene, and of each value, we test it on 100 random scenes as shown in Table.1. The following Fig.4 shows the tuning process of parameter ∂.

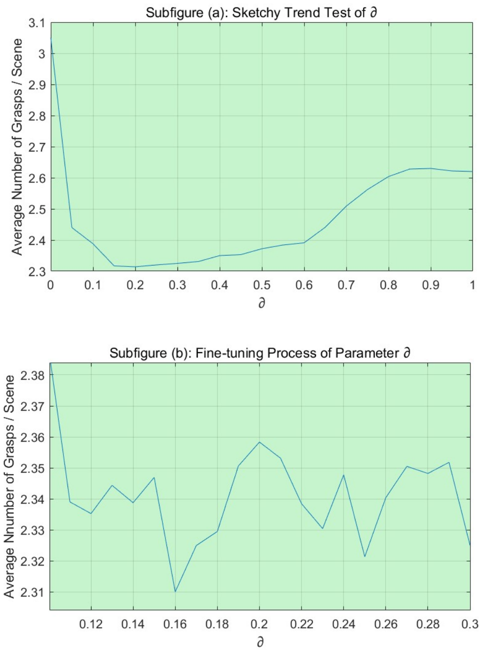

Fig.4 The Finetuning of Parameter ∂. ∂ is the lower limit in Eq.4, and this is critical in deciding whether it is necessary to grasp an object with a small occlusion ratio O. Subfigure (a) indicates the overall trend of the average number of grasps when changing ∂; subfigure (b) shows the finetuning of ∂ within the selected range 0.1~0.3.

We believe that in the process of deciding a grasp-worthy occluder, there is a threshold value ∂ related to the occlusion ratio that is critical to the grasp efficiency. On the one hand, if ∂ is too large, our decision algorithm will not grasp occluders which has already occluded a big area. On the other hand, a too small ∂ means grasping small occluders which do not actually preclude the process of grasping the target. To finetune the value ∂, we divided the test epochs into 3:2. The first part occupying $\frac{3}{5}$ of the dataset was used for tuning value ∂, and the second part was used to test the universality of our tuned model.

The average grasps per scene is the average grasp movements the robot completes until finally grasping the target object, including the grasps in de-occlusion procedure. As Fig.4 indicates, when ∂ is too small, the robot will be prone to grasp redundant objects, resulting in the greater number of grasps per scene; but when the threshold ∂ is assigned a too big value, it will lead to grasps directly to the target object with poor robustness. After finetuning threshold ∂ in a selected range, we yield the minimum grasps per scene 2.31 at the ∂ value of 0.16.

D. Grasp Efficiency Test Results

TABLE 2 GRASP EFFICIENCY TEST RESULTS

Average Success Rate of Grasping the Occluded Object **(Before and After De-occlusion)**	Before	**After**
	21.7%	**98.4%**
Average Grasp Movements per Scene	2.31	
Average Grasp Movements per Occluder	1.29	

We tested our model based on 10 test epochs indicated in Table.1. We noted the average success rate of grasping the occluded target object without and with the aid of our decision-making algorithm and BCNet occlusion prediction. And we yield 98.4% success rate after removing necessary occluders, which is around 4.5 times of the success rate without de-occlusion movements. This feature is particularly important in facilitating the overall grasping speed in real-world cluttered scenes. Under a cluttered test scene with up to 10 objects (indicated in Table.1), our average grasp movements in each scene is 2.31, which means we are mostly grasping necessary objects, instead of re-arranging the whole scene which involves a great number of redundant grasping.

V. Discussion

In our model, we combined two novel networks together using our decision-making algorithm to decide the next best grasp (NBG) in de-occlusion semantic grasping scenarios. The BCNet has endowed our model the ability to predict occlusion areas, which is an important criterion for determining the NBG. While making the decision, our model also considers the robustness of grasps, and we extract the grasp robustness from GraspNet. We finetuned our model parameters in the CoppeliaSim virtual environment and harvested considerable grasp efficiency. Compared to other grasp planning methods, our model does not require the robot to re-arrange or remove all the objects in the scene, which shortcuts redundant movements. Also, since our model is worked on a single camera viewpoint, it does not need exhausting observation over the whole grasp scene. However, our model still faces several challenges. Our model is not able to detect more than two occluding objects and is not fully capable of understanding occlusion from the above. We consider this model can better understand the spatial relationship of objects by combining the point cloud data with the instance segmentation masks. We are hoping to conduct further researches to better consummate our model.

VI. Conclusion

In our research, we are aimed at enhancing the grasp efficiency in planning to grasp occluded objects. Before grasping the target object, the robot is required to grasp and remove the occluding objects in order to create convenience for grasp pose detection on the target object. In the process of planning to grasp and remove occluding objects (occluders), we propose a decision-making algorithm to determine the next best grasp (NBG). Our decision-making algorithm considers both the robustness of a proposed grasp pose and the necessity of grasping one occluding object. To integrate BCNet and GraspNet which take care of occlusion prediction and grasp pose detection respectively, we conduct a mapping process which transfers the spatial grasp pose location to the image planar where the objects are segmented. At last, we finetune and test our model on the simulated grasp environment and harvest great grasp efficiency enhancement. Our work is greatly inspired by the real-world grasping obstacles that the robots are faced with. And we look forward that our work provides little reference for further studies on grasp planning in dense clutters, especially those scenarios where the targets are deeply occluded.

References

[1] Bohg, J., Hausman, K., Sankaran, B., Brock, O., Kragic, D., Schaal, S., & Sukhatme, G. S. J. I. T. o. R. (2017). Interactive perception: Leveraging action in perception and perception in action. *33*(6), 1273-1291.

[2] Du, G., Wang, K., Lian, S., & Zhao, K. J. A. I. R. (2021). Vision-based robotic grasping from object localization, object pose estimation to grasp estimation for parallel grippers: a review. 54(3), 1677-1734.

[3] Murali, P. K., Dutta, A., Gentner, M., Burdet, E., Dahiya, R., Kaboli, M. J. I. R., & Letters, A. (2022). Active visuo-tactile interactive robotic perception for accurate object pose estimation in dense clutter. 7(2), 4686-4693.

[4] Fang, H.-S., Wang, C., Gou, M., & Lu, C. (2020). Graspnet-1billion: A large-scale benchmark for general object grasping. Paper presented at the Proceedings of the IEEE/CVF conference on computer vision and pattern recognition.

[5] Ke, L., Tai, Y.-W., & Tang, C.-K. (2021). Deep occlusion-aware instance segmentation with overlapping bilayers. Paper presented at the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.

[6] Du, G., Wang, K., Lian, S., & Zhao, K. J. A. I. R. (2021). Vision-based robotic grasping from object localization, object pose estimation to grasp estimation for parallel grippers: a review. 54(3), 1677-1734.

[7] Redmon, J., & Angelova, A. (2015). Real-time grasp detection using convolutional neural networks. Paper presented at the 2015 IEEE international conference on robotics and automation (ICRA).

[8] Zhang, Q., Qu, D., Xu, F., & Zou, F. (2017). Robust robot grasp detection in multimodal fusion. Paper presented at the MATEC Web of Conferences.

[9] Gualtieri, M., Ten Pas, A., Saenko, K., & Platt, R. (2016). High precision grasp pose detection in dense clutter. Paper presented at the 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS).

[10] Chen, X., & Yuille, A. L. (2015). Parsing occluded people by flexible compositions. Paper presented at the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

[11] Chen, Y.-T., Liu, X., & Yang, M.-H. (2015). Multi-instance object segmentation with occlusion handling. Paper presented at the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

[12] Zhan, X., Pan, X., Dai, B., Liu, Z., Lin, D., & Loy, C. C. (2020). Self-supervised scene de-occlusion. Paper presented at the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition.

[13] Qi, C. R., Yi, L., Su, H., & Guibas, L. J. J. A. i. n. i. p. s. (2017). Pointnet++: Deep hierarchical feature learning on point sets in a metric space. 30.