Proceedings of the 2nd WRC Symposium
on Advanced Robotics and Automation 2019
Beijing, China, August 21, 2019

Predicting Grasping Order in Clutter Environment by Using Both Color Image and Points Cloud

Peichen Wu[1], Wenbo Chen[2], Hongrui Liu[1], Yifan Duan[1], Nan Lin[1], Xiaoping Chen[1]

Abstract— **Grasping selection for a individual object has been developed many years, and there many feasible proposals have been presented. However, for a robot serving in daily life, this capacity is far from enough. Because the scene in our common life is usually chaotic, where objects are often mutual influences, front and back occlusion, stack up and down. In this paper, we mainly focus on grasping plan and selection in this clutter scene. The first step of this work is to segment the objects from the input picture which are going to manipulate. We use the Faster R-CNN model to segment objects based on the color image for this model have a fine ability to detect multi-objects. For planning the correct grasping order in a chaotic scene, however, it is not enough only using the color information. So, we should combine the geometry information of point cloud together. In this process, we use the extrinsic parameters to transform the bounding-box of objects in color image to point cloud. Then, we calculate the grasping order whit the geometry information of point cloud. In experiment, we demonstrate the whole process of our proposal. And we actually grasping common objects in a clutter environment with the KINOVA ARM and an underactuated hand designed by ourselves.**

I. INTRODUCTION

Grasping is a basic and important ability for service robots.Since the end of last century, there have been many researchers engaging in related works. At present, for the grasping task of individual objects, there is already a mature solution. However, robotic manipulation of objects in clutter still remains a challenging problem.

The home environment in which service robots work is often a messy environment full of uncertainty. Therefore, the ability to pick and operate objects in cluttered environments limits the application of service robots to a large extent.

Objects in cluttered environments often have occlusions and stacking, which makes it more difficult to segment individual objects. In addition, objects in cluttered environments are not independent of each other, often contact each other, and even depend on each other. Therefore, it is necessary to plan a reasonable grasp order in the cluttered environment, so that the task can be completed without causing any damage to the objects.

In the light of the common clutter scene in daily life is shown in Fig.1, there are mutual influences, front and back occlusion, stack up and down. In this paper, we propose a reasonably planned object capture order, which can be successfully and safely used in chaotic environment.

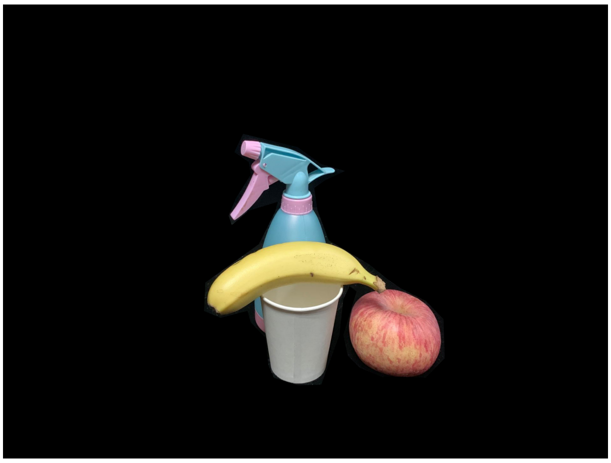

Fig. 1. Common objects are mutual influences, front and back occlusion, stack up and down. This scene often appears in daily life.

The main idea of our proposal is segmenting objects from clutter environment using color image and calculating the correct grasping order based on the geometry information of point cloud. So, we need a RGB-D camera which can provide both color image and point cloud. Moreover, we also need the extrinsic parameters to relate the color image and the point cloud.

II. RELATED WORK

Generally speaking, robot automatic grasp for a single object can be divided into two phases: grasp selection and grasp execution [1]. In grasp selection, the robot is given an object and need to find a stable grip position and posture. In grasp execution, the hand claw has reached the selected target position, then perform claw closure, stably and safely grab the object.

Grasp selection has been widely explored in recent decades. Miller et al [2] uses shape primitives for grasping. However, this method requires known and accurate object information. Geidensatam et al [3] uses the 3D point cloud feature of box-based to calculate 2D capture strategy. This approach requires a 3D model of a known object, which preventing it from being applied to the unstructured environment. In the work of [4] [5], the author uses the method of constructing a 3D model of objects to capture, the quality of these models determines the quality of grasping. Kaijien et al proposes a method of grasp selection based only on partial shape information, and design it to make adjustments

*This work is supported by the National Natural Science Foundation of China under grant U1613216 and 61573333.

[1] Peichen Wu, Hongrui Liu, Yifan Duan, Nan Lin and Xiaoping Chen are with the School of Computer Science and Technology, University of Science and Technology of China, Hefei, 230026, China.

[2] Wenbo Chen is with the School of Information Science and Technology, University of Science and Technology of China, Hefei, 230026, China. wpc16@mail.ustc.edu.cn and xpchen@ustc.edu.cn

978-1-7281-5552-4/19/$31.00 ©2019 IEEE

in according to the touch sensors on the finger in the process of grasp execution. In our previous work, we also proposed a grasp selection method based on partial visual information to calculate the shape change trend of objects [6]. All of these methods, however, assume that there is only one object in the scene, without considering the grasp selection in a messy environment.

Mehmet et al proposed a framework of Push-grasping in chaotic environment [7]. The author uses the action library containing four basic actions to complete the target task by rearranging. In [8] [9], the author also accomplishes the task by using the motion planning method of pushing based in the chaotic environment. Haustein et al used kinodynamic-RRT method to avoid Expensive planning in clutter configuration space [10].

Weihao et al [11] based on reinforcement learning method rearrange objects in scene, so as to realize manipulation task in cluttered environment. Wissam et al proposed a motion planning with a receding horizon for manipulation using a learned value function to realize closed in clutter environment [12]. In [13], the author proposes a real-time online re-planning under clutter algorithm to deal with the uncertainty caused by the rearrange process. However, these methods usually need to know the position and shape of obstacles. What's more, there is no mutual support or occlusion between objects though the objects in the scene are messy.

In recent years, with the development of in-depth learning, some researchers have used deep neural networks to estimate the 6D attitude information of objects in the scene and then implement the grasp task. Xiang et al proposed a convolution neural network for 6D object pose estimation by combining RGB and depth information of the object [14]. Based on that, C Hen et al proposed a dense fusion network which fuses RGB and depth information Pixel-wise [15]. This kind of method needs to label 6D information of a large number of objects, which is a heavy task in reality. Moreover, this kind of method does not consider the problem of mutual supports between objects.

In [16], the author proposes a learning-based method to predict the reasonable and safe grasp order under the condition that objects are supported by each other in a chaotic environment. This approach depends on elaborate hand-craft futures to predict the grasp order. And the data set that is already marked is needed to improve the accuracy of the prediction grasp order.

However, in a clutter environment before selecting grasping position, there should be another step aimed at segmenting object from each other. In total, there have two prime method. One segments objects with color image and the other one uses points cloud. After that, calculating a correct grasping order according to the information of objects as more as we can obtain is the next step. Once we get the grasping order, we then select the grasping position for the current target object.

In this paper, we propose a new method for planning the grabbing order of objects in chaotic environments and in the case of interaction between objects. First, we use faster R-CNN [17] to segment objects in RGB images. Then, we predict the grab order based on the relationship between the point cloud information and the object bounding box.

III. Models and Mechanical Analysis

The first step for grasping in a cluttered environment, is segmenting objects from each other. Many proposals have been presented to realize this by using color image or points cloud. In this section, we introduction our segmentation using Faster R-CNN in color image firstly. Then, we transform the bounding-boxes from optical frame to depth frame to prepare for calculating the grasping order. Finally, we present the method to calculate grasping order by using the geometry information in points cloud.

A. Color Image Segmentation Using Faster R-CNN

Faster R-CNN is a multi-objects recognition model, it performs well in PASCAL VOC 2007 and MS COCO data sets. Fig. 2 shows the structure of the Faster R-CNN model. The feature extraction network is followed by a region proposal network (RPN). A window of size n × n slides onto the feature map, and at each location it stays the features in the window are mapped to a low-dimensional vector, which will be used for object-background classification and proposal regression. At the same time, k region proposals centered on the sliding window in the original image are extracted according to k anchors, which are rectangular boxes of different shapes and sizes. Moreover, for each proposal, two probabilities for the classification and four parameters for the regression will be achieved, composing the final 6k outputs of the classification layer and the regression layer. The sliding window, classification layer and regression layer are all implemented using convolutional neural networks.

The input for the Faster R-CNN model is a color image. And after the network processing, we will obtain the objects which are going to grasping with their bounding-boxes in the optical frame. We use b_i^c to denote the value of a bounding-box in the optical frame. So, the set of bounding-boxes in the optical frame is defined as B^c.

$$b_i^o \in B^c \tag{1}$$

Where $i = 0, 1, 2...N$, meaning there are N objects in the input image.

B. Transforming Bounding-boxes from Optical Frame to Depth Frame

As shown in Fig. 3, a RGB-D camera usually has two lens to obtain color image and points cloud. The color image is based on the optical frame while the points cloud is based on the depth frame. Using the extrinsic parameters between the two lens, we can transform values in optical frame to depth. T_o^d is a homogeneous transformation matrix, which contains the rotation and offset parameters.

In the last subsection, we have got a set of bounding-boxes in the optical frame after inputting a color image. Our aim is to calculate a right order to grasp in a clutter environment. In this scene, some objects may occlude others. And sometimes,

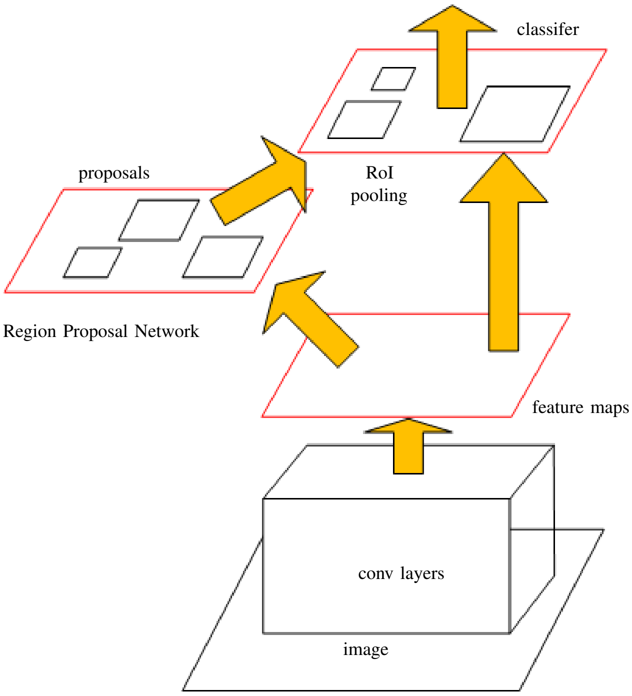

Fig. 2. Flowchart of Faster R-CNN. We use the Faster R-CNN model to segment objects with color image. The output of the model is the label of object and corresponding bounding-box.

they might even be stacked. So, we need to combine more information to achieve our purpose beside color information. Base on this, we then transform the bounding-boxes from optical frame to depth frame.

$$\begin{aligned} b_i^d &= T_o^d \cdot b_i^o \\ \wedge\, b_i^o &\in B^c \end{aligned} \tag{2}$$

Where b_i^d is the value of a bounding-box in the depth frame. We can get bounding-boxes of object in points cloud through Equation 2. Correspondingly, we have a bounding-box set, B_i^d.

$$b_i^d \in B_i^d \tag{3}$$

When having B_i^d, we can obtain the geometry information align the specific object. Then, we will calculate the suitable grasping order based these information.

C. Calculating grasping order

This section introduces a proposal of point cloud-based object sorting and grab order selection algorithms and the full flow of grabbing objects.The overall goal is to determine the order in which all objects in the field of view are properly captured, and also to determine the order in which a particular target object is captured.

The principle of total grabbing is to grasp objects in front of the object, without any other objects, that is, objects that will not affect other objects in the environment. The three directions are left, top, and front in order of priority. Image information is collected in a frontal view with a certain angle of depression. The whole process is shown in Fig 4.

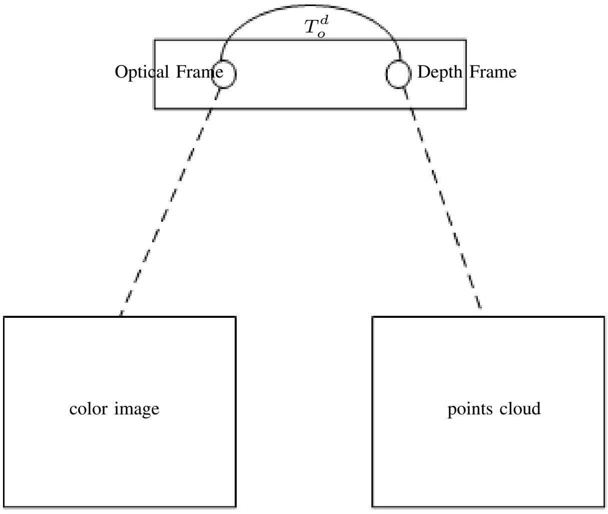

Fig. 3. A general schematic of a RGB-D camera. A camera usually has two lens to obtain color image and points cloud. Extrinsic parameters can transform one of them to the other.

1) Construct a structure sequence: After obtaining the point cloud information, we firstly filter out the bottom surface. Since the background point cloud depth value is always larger than the object point cloud information, it will not affect the crawling in the subsequent sorting process and may not be considered. Then, using the clustering algorithm, the item structure is used to store the highest value information of the upper and lower left and right borders of the point cloud of the object and the calculated xyz coordinate of the center line of the line of sight(x, y, z respectively represent the left and right distance, the depth of field and the height of the upper and lower sides as shown in Fig 4). And then create a sequence with the object structure as an element $S = object_1...object_n$, where $object_n = maxup, down, left, Right, coordinatex, y, z$.

2) Left and right relationship determination: Traverse the structure sequence, sorting S_1 according to the left boundary value of each element from small to large, and deleting all the elements of $max.left < object_1.right$ with the right boundary value of the first element as the standard.It can be judged that all the objects satisfying the above conditions are on the right side of the left boundary object, and have no upper and lower support relationship, so the deletion does not affect the selection of the grab target.

3) Determining context: Traverse the sequence of structures, sorting S_2 according to the point y coordinate of the point cloud center point from small to large. Since this step is sorted, the y coordinate of the background point cloud must be larger than the point cloud of any object, so there is no need to filter out the background when initially processing the point cloud information.

4) Determine the lowest front object: Move the bottom element's smallest element to the first position of the sequence. This step is intended to prevent the upper and lower stacks from causing the object to be detected to be lower than the high point of the object.

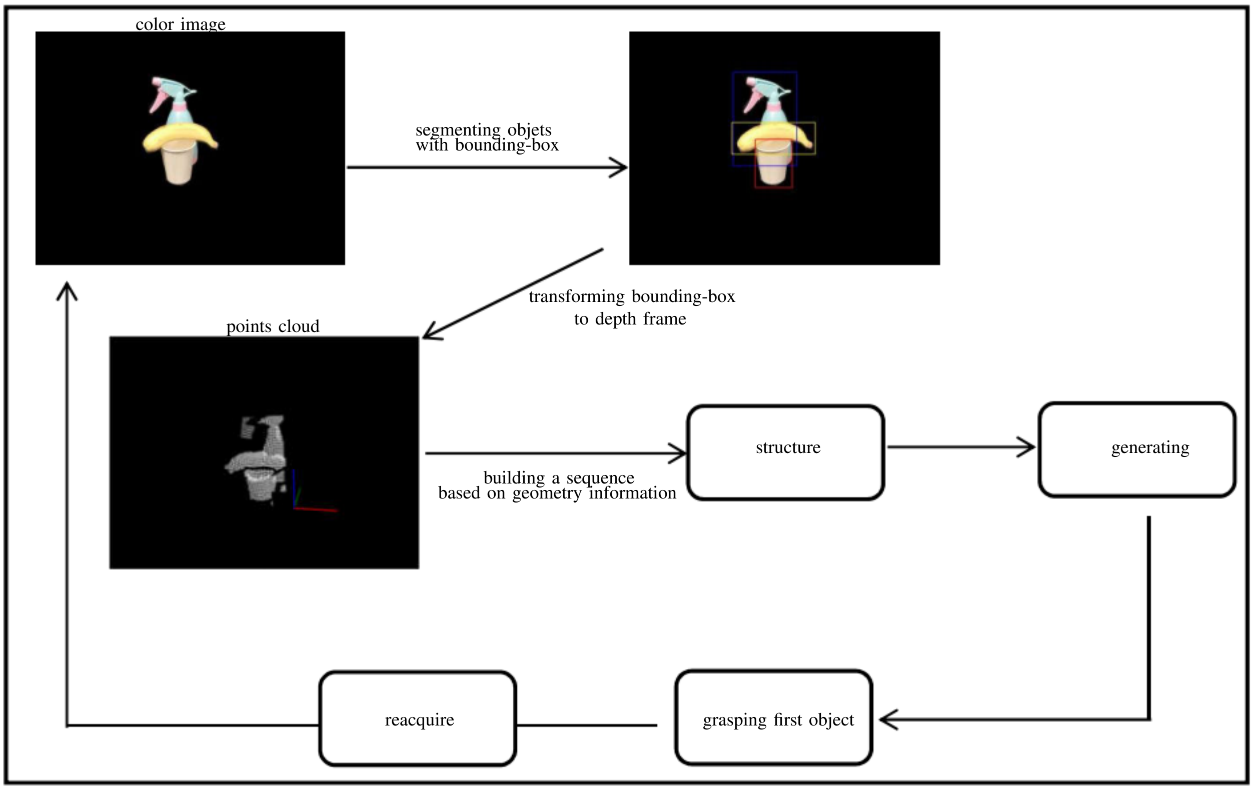

Fig. 4. A flow diagram of our method for calculating grasping order. In points cloud picture, the x, y, z axes are corresponding to the red, green, blue axes respectively.

5) Sort up and down: Traversing from the second bit of the sequence. If $object_k$ is the one above $object_1$, we modify $object_k$(down) to the boundary value of the point cloud at the upper and lower boundary between $object_k$ and $object_1$. This step is to prevent the object from being lower than the lowest point of the point cloud due to the special shape. The highest point of the object below, which causes an error in the process of judging the relationship. If $object_k(down) \geqslant object_1(high) - \varepsilon(\varepsilon << object_1(high) - object_1(down))$, this advances $object_k$ to be the first place in the sequence. Then we have a S_3 sequence, which is the final grasping sequence.

Under normal circumstances, after the above procedure, the first item of the S_3 sequence can be obtained as the top left front object, and the grabbing of it will not affect other objects in the environment. However, when the front and rear objects are in close proximity and the split angle cannot be divided, the lower boundary of the object supported above the rear object will be lower than the lowest front object, so it will be advanced to the first position. It also has no effect on the crawling, so the above algorithm is considered feasible. At this point, the first element of the S_3 sequence is the object should be grasped firstly.

Then, we will use the method in [6] to select grasping point. The method is our previous work. It works well even we only get a part of points cloud. It is designed for having adaptive capacity gripper, the cost time for it is less. The method mainly calculates grasping position by using the absolute maximum of the second derivative of the function from the front to the end of one object.

Algorithm 1 Calculate the grasping order

```
while Input image has target object do
    Segmenting color image with bounding-box
    Transforming bounding-box to depth frame
    Acquiring points cloud of objects based on transformed bounding-box
    if The points cloud are not an empty set then
        $S$=Create sequence (cluster (effective point cloud))
        $S_1$=left and right sorting (S)
        $S_2$=before and after sorting $(S_1)$
        Finding the smallest element of the lower boundary to be the first place in the sequence
        $S_3$=up and down sorting $(S_2)$
        Calculating grasping position for the first place object in $S_3$
    end if
end while
```

IV. Experiment

In this section, we demonstrate our whole model for planning and grasping objects in a clutter scene during actually grasping based on a hand-eye platform. In this scene, the objects are occluded front and back, stack up and down. The objects used are daily living.

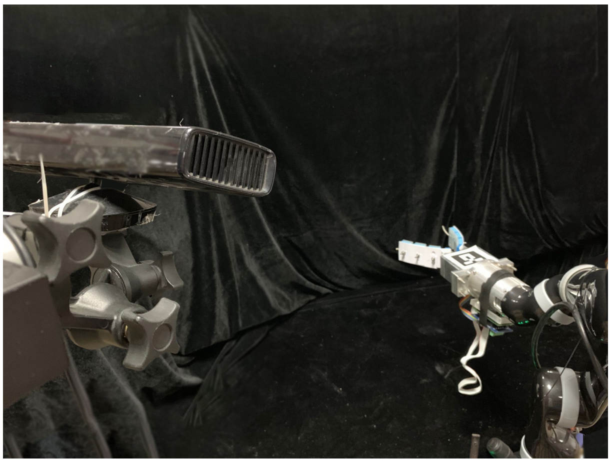

Fig. 5. The hand-eye platform for actually grasping objects.

A. Setup

In order to grasp objects in a clutter scene actually, we build a hand-eye grasping platform as shown in Fig. 5. We use a RGB-D camera (KINECT ONE, MICROSOFT) to obtain both color image and point cloud. And we transform bounding-boxes from optical frame to depth frame with the default extrinsic parameters provide by the manufacturer. The gripper is an underactuated one designed by ourself [18]. The carrier of the gripper is a six degree of freedoms arm(MICO, KINOVA) [19]. We calibrate the hand-eye platform using the method in [20]. We use the MICROSOFT COCO data [21] set to train the Faster R-CNN model. The data set has eighty categories of common objects. It almost covers everyday objects.

B. Actually grasping

In this section, we will demonstrate a entire process for a actual grasping in a clutter scene. In this scene, there have one object occluding another one and one object stacking another one. The left column in Fig. 6 is the object that should be grasped first after calculating the grasping order. The middle column demonstrates the grasping position for the target object calculating by our previous work. The red points represent the position of gripper. And the right column is the picture of the final execution result.

As we can see, the banana in Fig. 6 is the top and front object, so it is the first one to be grasped. After that, the cup is in front of the bottle. The cup is the next to be grasped. Finally, there is only a bottle on the platform, so just grasping it. The entire process is within 0.5 second, and our softwares are all run on ROS(Robot Operating System).

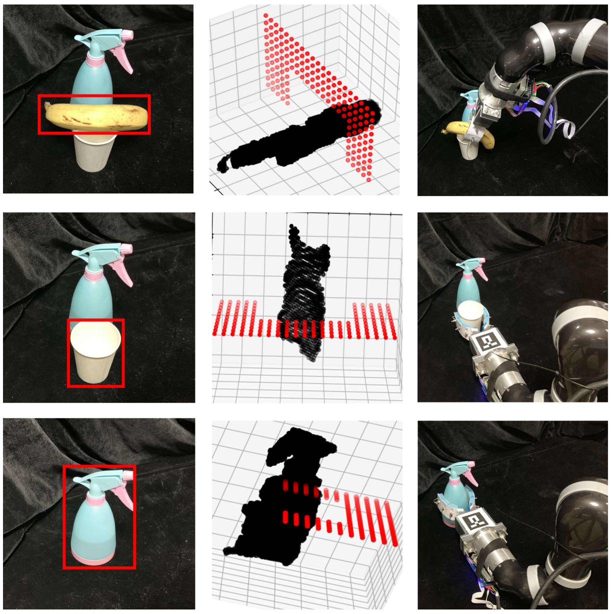

Fig. 6. The entire process for actually grasping. The left column is the target object going to grasp within the red bounding-box. The red points in middle column picture show the gripper position to grasp. And the right column is the picture after grasping target object.

V. Conclusion

In this paper, we realize a method aimed at calculating the correct grasping order in a clutter environment, by combining color image and geometry information in points cloud. Firstly, we segment objects from input color image with bounding-box by using Faster R-CNN model. Then we transform the bounding-box from optical frame to depth frame with the extrinsic parameters of RGB-D camera in order to obtain the geometry information of target objects. Finally, we calculate the grasping order. Besides, we also actually grasping chaotic objects on a calibrated hand-eye platform.

There is room for improvement in the present method. Using bounding-box to segment objects will involve some points cloud information, when there is another object in front of the target object. This raises the difficulty while calculating grasping order. In the future, we want to use some pixel-segmentation method to segment objects. Moreover, we are going to look at more complicated situations in next step, for example there is a object containing another.

References

[1] K. Hsiao, S. Chitta, M. Ciocarlie, and E. G. Jones, "Contact-reactive grasping of objects with partial shape information," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2010, pp. 1228–1235.

[2] A. T. Miller, S. Knoop, H. I. Christensen, and P. K. Allen, "Automatic grasp planning using shape primitives," 2003.

[3] S. Srinivasa, D. I. Ferguson, M. V. Weghe, R. Diankov, D. Berenson, C. Helfrich, and H. Strasdat, "The robotic busboy: Steps towards developing a mobile robotic home assistant," 2008.

[4] Z.-C. Marton, L. Goron, R. B. Rusu, and M. Beetz, "Reconstruction and verification of 3d object models for grasping," in *Robotics Research*. Springer, 2011, pp. 315–328.

[5] Z. C. Marton, R. B. Rusu, D. Jain, U. Klank, and M. Beetz, "Probabilistic categorization of kitchen objects in table settings with a composite sensor." in *IROS*, 2009, pp. 4777–4784.

[6] P. Wu, N. Lin, Y. Duan, T. Lei, L. Chai, and X. Chen, "An automatic grasp system with sensor feedback based on soft gripper," in *2018 WRC Symposium on Advanced Robotics and Automation (WRC SARA)*. IEEE, 2018, pp. 1–7.

[7] M. Dogar and S. Srinivasa, "A framework for push-grasping in clutter," *Robotics: Science and systems VII*, vol. 1, 2011.

[8] N. Kitaev, I. Mordatch, S. Patil, and P. Abbeel, “Physics-based trajectory optimization for grasping in cluttered environments,” in *2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2015, pp. 3102–3109.

[9] J. E. King, J. A. Haustein, S. S. Srinivasa, and T. Asfour, “Nonprehensile whole arm rearrangement planning on physics manifolds,” in *2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2015, pp. 2508–2515.

[10] J. A. Haustein, J. King, S. S. Srinivasa, and T. Asfour, “Kinodynamic randomized rearrangement planning via dynamic transitions between statically stable states,” in *2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2015, pp. 3075–3082.

[11] W. Yuan, J. A. Stork, D. Kragic, M. Y. Wang, and K. Hang, “Rearrangement with nonprehensile manipulation using deep reinforcement learning,” in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 270–277.

[12] W. Bejjani, R. Papallas, M. Leonetti, and M. R. Dogar, “Planning with a receding horizon for manipulation in clutter using a learned value function,” in *2018 IEEE-RAS 18th International Conference on Humanoid Robots (Humanoids)*. IEEE, 2018, pp. 1–9.

[13] W. C. Agboh and M. R. Dogar, “Real-time online re-planning for grasping under clutter and uncertainty,” in *2018 IEEE-RAS 18th International Conference on Humanoid Robots (Humanoids)*. IEEE, 2018, pp. 1–8.

[14] Y. Xiang, T. Schmidt, V. Narayanan, and D. Fox, “Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes,” *arXiv preprint arXiv:1711.00199*, 2017.

[15] C. Wang, D. Xu, Y. Zhu, R. Martín-Martín, C. Lu, L. Fei-Fei, and S. Savarese, “Densefusion: 6d object pose estimation by iterative dense fusion,” *arXiv preprint arXiv:1901.04780*, 2019.

[16] S. Panda, A. A. Hafez, and C. Jawahar, “Single and multiple view support order prediction in clutter for manipulation,” *Journal of Intelligent & Robotic Systems*, vol. 83, no. 2, pp. 179–203, 2016.

[17] S. Ren, K. He, R. Girshick, and J. Sun, “Faster r-cnn: Towards real-time object detection with region proposal networks,” in *Advances in neural information processing systems*, 2015, pp. 91–99.

[18] N. Lin, P. Wu, X. Tan, J. Zhu, Z. Guo, X. Qi, and X. Chen, “Design and analysis of a novel sucked-type underactuated hand with multiple grasping modes,” in *International Conference on Robot Intelligence Technology and Applications*. Springer, 2017, pp. 299–312.

[19] Kinova robotics. [Online]. Available: https://www.kinovarobotics.com/en

[20] R. Y. Tsai and R. K. Lenz, “A new technique for fully autonomous and efficient 3d robotics hand/eye calibration,” *IEEE Transactions on robotics and automation*, vol. 5, no. 3, pp. 345–358, 1989.

[21] Microsoft coco data set. [Online]. Available: http://cocodataset.org/#home