applied sciences

MDPI

Article

Semantic 3D Reconstruction for Robotic Manipulators with an Eye-In-Hand Vision System

Fusheng Zha [1], Yu Fu [1], Pengfei Wang [1], Wei Guo [1], Mantian Li [1,2,*], Xin Wang [2,*] and Hegao Cai [1]

1 State Key Laboratory of Robotics and System, Harbin Institute of Technology, Harbin 150080, China; zhafusheng@hit.edu.cn (F.Z.); 6120810528@hit.edu.cn (Y.F.); wangpengfei@hit.edu.cn (P.W.); wguo01@hit.edu.cn (W.G.); zfsh751228@163.com (H.C.)

2 Shenzhen Academy of Aerospace Technology, Shenzhen 518057, China

* Correspondence: limt@hit.edu.cn (M.L.); xin.wang@chinasaat.com (X.W.)

Received: 16 December 2019; Accepted: 3 February 2020; Published: 10 February 2020

Abstract: Three-dimensional reconstruction and semantic understandings have attracted extensive attention in recent years. However, current reconstruction techniques mainly target large-scale scenes, such as an indoor environment or automatic self-driving cars. There are few studies on small-scale and high-precision scene reconstruction for manipulator operation, which plays an essential role in the decision-making and intelligent control system. In this paper, a group of images captured from an eye-in-hand vision system carried on a robotic manipulator are segmented by deep learning and geometric features and create a semantic 3D reconstruction using a map stitching method. The results demonstrate that the quality of segmented images and the precision of semantic 3D reconstruction are effectively improved by our method.

Keywords: semantic 3D reconstruction; eye-in-hand vision system; robotic manipulator

1. Introduction

In an unstructured environment, the type and shape of the objects are unpredictable. While, in order to achieve autonomous operations, the robot must be able to use visual sensors, such as lasers or cameras, to get the information about the scene [1–3]. Therefore, the robot can obtain features and identify relevant objects in the surrounding environment and then plan the motion accordingly. In the process, besides providing the location information of objects, a semantic 3D map can facilitate its decision-making based on actual world processes, such as judging the stability of the scene objects [4–6], grasping and placing objects by imitating human beings [7], and generating relevant action sequences [8–10].

Environmental information is usually collected by different sensors, such as lasers [11], a monocular camera [12], or a depth camera [13], and is then processed through a series of algorithms, such as height estimation [14,15], target detection, image segmentation, visual odometer, and image stitching to generate an environmental map, which is called simultaneous localization and mapping (SLAM) or structure from motion (SFM). The visual odometer-based method seriously affects the accuracy of the mapping due to the position error caused by the sensors. However, the eye-in-hand vision system is more accurate than the visual odometer. Therefore, it is necessary to make full use of the high accuracy of the robotic manipulator to improve the quality of the 3D reconstruction of the scene [16,17]. Another problem is that the precision of semantic segmentation is still insufficient, even by the latest method, so it is necessary to find a way to improve the quality of semantic segmentation.

Therefore, we explore to establish an integrated 3D object semantic reconstruction framework for eye-in-hand manipulators, including RGBD image segmentation, camera pose optimization, and map

stitching. This enables us to achieve the following: (1) combine deep learning with geometric feature methods to perform the semantic segmentation; (2) employ the object point cloud segmentation-based Segment Iterative Closest Point (SICP) method to optimize the camera pose and position; and (3) stitch together a semantic 3D map by data association.

In summary, the main contributions of this work are:

- The accuracy of image segmentation and the quality of object modeling are improved with an eye-in-hand manipulator through combining deep learning with geometric methods.
- A high-precision semantic 3D map is established by applying the SICP method to optimize the camera position.

The paper is organized as follows: related works and the present work are described in Sections 2 and 3, respectively. In Section 4, the experimental results are detailed and presented. The discussion and conclusion are given in Section 5.

2. Related Works

As previous 3D reconstruction using an eye-in-hand camera rarely contains semantic information and, currently, a large number of semantic 3D reconstruction is based on hand-held cameras, we discuss the following two parts: semantic 3D reconstruction based on an eye-in-hand camera and a hand-held camera.

2.1. Semantic 3D Reconstruction Based on an Eye-in-Hand Camera

Since the position of the object in the 3D space is necessary for robotic manipulators to operate objects, the eye-in-hand camera is usually applied to get this information and make 3D scene reconstruction. Fuchs et al. [18] used Time of Flight (ToF) cameras to acquire images and optimize the images through the Iterative Closest Point (ICP) algorithm. Barth et al. [19] used the LSD-SLAM method to create sparse scene maps, using object edge information to identify objects. Chang et al. [20] used a monocular eye-in-hand camera and a laser radar to obtain the point cloud of the scene and combined it with the Computer Closer Point (CCP) and ICP methods to improve the matching accuracy. The above methods can only build 3D maps without semantic information, causing them have to use all the point clouds to perform ICP matching. That induced a low calculation speed and low matching precision due to the background interference. Moreover, since there is no semantic segmentation of the scene, the object-level 3D reconstruction cannot be achieved.

2.2. Semantic 3D Reconstruction Based on a Hand-Held Camera

After years of development, 3D scene reconstruction based on vision has been relatively mature and has produced a large number of excellent algorithms [21–23]. With the improvement of target detection and image segmentation algorithms, semantic 3D scene reconstruction has become a research hotspot in recent years [24–29]. Its essence is the effective combination of semantic information with the SLAM system to generate 3D maps with semantic labels. Single Shot Detectors (SSDs) are introduced to handle geometric feather-based point cloud segmentation on the foundation of the orb-slam and processed map fusion through data association and the ICP [30]. Based on probabilistic methods, lots of previous works conduct 2D image segmentation through Random Decision Forests (RDF) and integrate 2D image labels into a 3D reconstruction map with a conditional random field and Bayesian updating model [31]. McCormac et al. [32] used the Convolutional Neural Network (CNN) to obtain the probability distribution of Classification for each 2D pixel, and then the Bayesian updating model would track the classification probability distribution of each curved surface, which would be updated based on the information regarding the data association provided by the SLAM system. In the subsequent work, they created a SLAM system with 6 degrees of freedom by merging 2D object detection with a 3D voxel foreground [33]. Bowman et al. [34] proposed a theoretical framework for the fusion of scale and semantic information, realizing the dynamic tracking of objects through ICP

and RGB error and achieving the real-time object 3D reconstruction by asynchronous frame updating. Although the above works have established an environmental semantic map, the map scale is usually too large to reach a high accuracy, which limits its application in elaborate 3D modeling, such as desktop objects. The aforementioned 3D reconstruction method commonly use a hand-held camera and need a visual odometer, while in the eye-in-hand vision system in robotic manipulators, the 3D reconstruction method can be simplified through a forward kinematics analysis of robotic manipulators.

3. Overview of the Proposed Method

Our algorithm includes fusion segmentation, combining deep learning with geometric feature methods, camera pose optimization, and map stitching. The algorithm flow is shown in Figure 1. The deep learning adopts the R-50-FPN structure of mask R-CNN, and the geometric feature method adopts supervoxel segmentation and the Locally Colored Convex Connected Patches (LCCCP) clustering method with color information. The fusion segmentation uses neural network segmentation results to further cluster LCCCP segmentation mass to generate a high-precision segmented point cloud with semantic information and then apply the split point cloud of two adjacent frames for ICP matching to get the real camera position. The segmented point cloud is transformed to the world coordinate system through the current real camera position, and the data association method based on the gravity center distance is adopted to judge whether the segmented point cloud is a false recognition. If there is no false recognition, the segmented point cloud is spliced in the map. A 3D model reconstruction of each object is realized by splicing the point cloud at different positions from multiple angles.

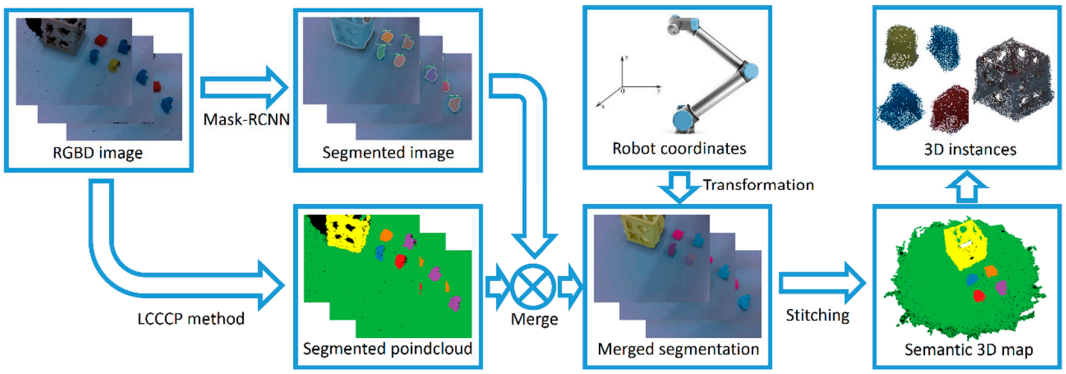


Figure 1. Overview of our method. This process is mainly divided into two parts: image segmentation and map stitching.

3.1. Object Recognition and Fusion Segmentation

The semantic segmentation algorithm is the basis of map stitching. Pictures and point clouds are segmented by neural networks and geometric features, respectively, and finally the two parts are fused together to generate semantic information. Therefore, this algorithm includes three parts: 2D semantic segmentation, point cloud segmentation, and semantic fusion.

3.1.1. Target Detection and Instance Segmentation Based on 2D Images

Among numerous methods for object detection and instance segmentation based on 2D images, see, e.g., [35–39], mask R-CNN is one of the most pragmatic instance segmentation frameworks at present, which can effectively detect objects in images and simultaneously generate a high-quality segmentation mask for each instance. Based on previous classification and regression branches in Faster-CNN, it adds another branch, which segment and output each region of interest (ROI) to achieve semantic segmentation [40]. The object recognition and 2D image segmentation in our work are constructed according to mask R-CNN framework.

3.1.2. Point Cloud Segmentation Based on the Geometric Feature Method

Although mask R-CNN has a relatively high recognition accuracy, the image segmentation accuracy is still insufficient, so it is difficult to achieve high-precision 3D reconstruction by merely adopting 2D image segmentation. In order to improve the accuracy of segmentation, we also take advantage of the 3D point cloud segmentation method. Firstly, the point clouds have been decomposing into many small patches by way of supervoxel segmentation to implement over-segmentation and then perform clustering analysis using the locally convex connected patches (LCCP) method [41].

The aforementioned LCCP method merely utilizes position and normal as not relying on the point cloud color. Suppose $\vec{p_i}$ and $\vec{p_j}$ represent two adjacent supervoxels, $couv\left(\vec{p_i}, \vec{p_j}\right)$ represents whether the connection between two supervoxels is convex. Extended Convexity Criterion and Sanity Criterion can be expressed with $\mathrm{CC}_e\left(\vec{p_i}, \vec{p_j}\right)$ and $SC\left(\vec{p_i}, \vec{p_j}\right)$, respectively [41].

Since the conventional LCCP method is not able to recognize two objects when the surface of different objects is tangential, it is necessary to differentiate objects by means of color information. In consideration of this problem, we improve the LCCP method by adding a parameter named the Point Color Criterion (PCC). We define γ as the maximum value of color-difference between two adjacent supervoxels, that is:

$$\gamma\left(\vec{p_i}, \vec{p_j}\right) = \max\left(\left|R_{\vec{p_i}} - R_{\vec{p_j}}\right|, \left|\mathrm{G}_{\vec{p_i}} - \mathrm{G}_{\vec{p_j}}\right|, \left|\mathrm{B}_{\vec{p_i}} - \mathrm{B}_{\vec{p_j}}\right|\right) \tag{1}$$

where $\gamma\left(\vec{p_i}, \vec{p_j}\right)$ is larger than the threshold value γ_{thresh}, the two supervoxels are recognized as two different objects. γ_{thresh} is an important parameter, which depends on the color difference between the objects. It is generally set to be a small value. Therefore, even if the color differences between the objects are small, the algorithm can also distinguish between them. However, too small a γ_{thresh} will cause over-segmentation. The color criterion of point cloud can be defined as:

$$\mathrm{PCC}\left(\vec{p_i}, \vec{p_j}\right) := \begin{cases} \text{true} & \gamma\left(\vec{p_i}, \vec{p_j}\right) < \gamma_{\text{thresh}} \\ \text{false} & \text{otherwise} \end{cases} \tag{2}$$

As a result, the LCCCP method is judged by the criteria:

$$\mathrm{conv}\left(\vec{p_i}, \vec{p_j}\right) = \mathrm{CC}_e\left(\vec{p_i}, \vec{p_j}\right) \wedge SC\left(\vec{p_i}, \vec{p_j}\right) \wedge \mathrm{PCC}\left(\vec{p_i}, \vec{p_j}\right) \tag{3}$$

3.1.3. Fusion Segmentation

As described above, the 2D image segmentation method relying on the neural network can segment multiple objects simultaneously with poor accuracy, while the geometric feature segmentation method is characterized by high edge accuracy but a tendency towards over-segmentation and a lack of semantic information in the segmented block. So, it is indispensable to combine the two methods to achieve a high-precision semantic instance segmentation. Assuming that 50% of the segmented patches generated by the LCCCP method are in the segmented image produced by mask R-CNN, the segmented block is marked as the object. Count all the segmented patches belonging to the object and merge them into the point cloud P_0^c of the current frame object in the camera coordinate system.

3.2. Camera Pose Optimization

Due to the motion error of the manipulator, the position of the eye-in-hand camera will deviate from the target position. If the point cloud of the current frame is directly spliced into the map, it will lead to point cloud model misalignment, so the registration method is necessary to be employed to optimize the camera pose and the SICP method is applied to calculate the camera pose deviation.

Supposing that the point cloud of the current frame in the camera coordinate system is P_0^c, the point cloud in the world coordinate system is P^w, the transform matrix at the end of the manipulator of the

current frame relative to the world coordinate system is $T_{w,t}$, the transform matrix of the camera relative to the end of manipulator is $T_{t,c}$, the transform matrix of the current frame in the world coordinate system is $T_{w,c} = T_{w,t}T_{t,c}$. After being transformed by $T_{w,c}$, the current frame object point cloud matches with the map point cloud by the SICP algorithm P^w to obtain the optimization transformation T_{ICP}, and the point cloud P_0^w of current frame object after compensation in the world coordinate system is:

$$P_0^w = T_{ICP}T_{w,t}T_{t,c}P_0^c \tag{4}$$

3.3. Data Association and Map Stitching

After transforming the point cloud of the current frame to the world coordinate system, it is essential to judge whether the transformed point cloud label is correct. Based on the previously reported method, the point cloud of the instance object in the world coordinate system is P_0^w. Assuming that there are m objects of the same category in the current map, we calculate the point cloud gravity center C_0 of each object point cloud $\left\{P_1^w, \cdots P_m^w\right\}$. The object point cloud P_t^w is:

$$P_t^w = arg\min_p \|C_i - C_0\| \tag{5}$$

Using this, we are able to calculate the Euclidean distance between all point pairs, which is from the current object point cloud P_0^w to the target object point cloud P_t^w. The value of ζ depends on the similarity between two sequential images. Parameter ζ usually takes a small value. The algorithm can identify semantic errors and avoid wrong splicing. However, we cannot set ζ too low, because when the similarity between two sequential images is poor, many segmentation results will be discarded. If more than 50% of the distance between point pairs is less than ζ, then the matching is considered successful, otherwise it is classified as a misidentification. Generally, this process takes $\zeta = 2$ mm. After successful matching, the object point cloud is merged into the point cloud map with voxel filtering.

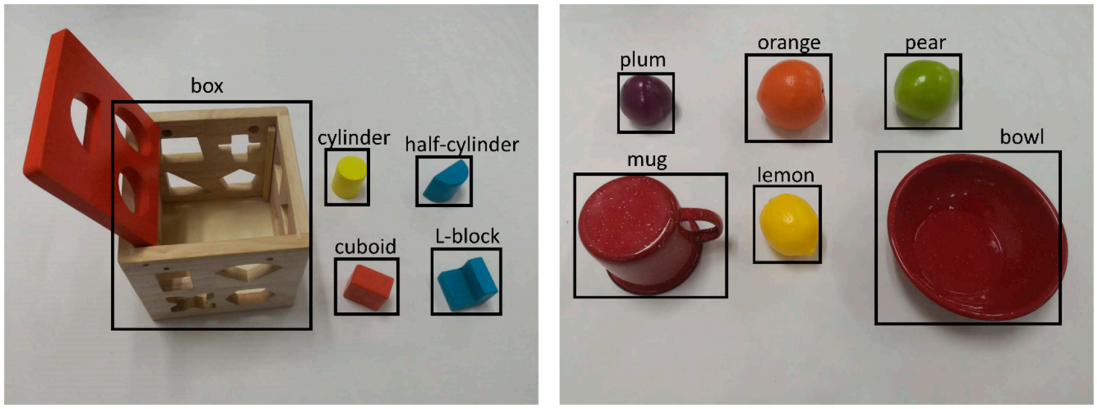
4. Experimental Results

To verify the precision and reliability of our algorithm, we completed a series of experiments on image segmentation and 3D reconstruction by a robotic manipulator with the eye-in-hand vision system. Each experiment has been repeated 10 times.

4.1. Experimental Conditions

We assembled a RealSense D435 camera at the end of UR10 robotic manipulator to take photos at 400 mm away from the desktop with resolution at 640 × 480 pixels. We controlled the robotic manipulator with an eye-in-hand camera system to take 16 pictures every 360 degrees around the object.

We validate our algorithm by employing two different datasets. Our dataset contains five types of toys, namely cylinder, half-cylinder, L-block, cuboid, and box, as shown in Figure 2a. The dataset has a total of 1200 images shot at different angles. The other dataset comes from the Yale-CMU-Berkeley (YCB) Benchmarks [42], which contain objects of different sizes and shapes in daily life. We chose lemon, pear, orange, plum, bowl, and mug for a total of six objects, as shown in Figure 2b.

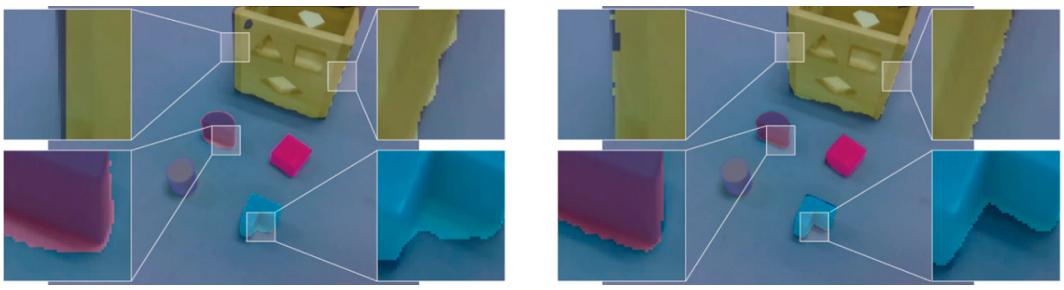


(**a**) (**b**)

Figure 2. Datasets in this paper. (**a**) The objects in our dataset, including the cylinder, half-cylinder, L-block, cuboid, and box, and (**b**) the objects in the Yale-CMU-Berkeley (YCB) Benchmarks, including lemon, pear, orange, plum, bowl, and mug.

4.2. Image Segmentation Results

The mask R-CNN adopts an R-50-FPN structure and is trained by 1200 manually labeled images with 5 types of objecting in the training set. The images are processed with instance segmentation according to the above method, and the segmentation result, which is shown in Figure 3, is compared with the mask R-CNN method. Figure 3a is the qualitative segmentation result of mask R-CNN. Each color represents one type of object. The edge of the segmented image is far from the edge of the actual object, and a hole may be generated in the segmentation area. Figure 3b is the segmentation result of our method. Because the geometric features at the edge of the object change drastically, while the geometric features of the object are stable, the image segmentation method based on geometric features makes the segmentation on the edge of the object more delicate with the segmentation edge closer to the real value and the segmentation region more complete.



(**a**) (**b**)

Figure 3. Comparison of the segmented image. (**a**) The segmentation results of mask R-CNN, and (**b**) the segmentation results of our method.

The quantitative comparison criteria is referred to in [43]. The Intersection-over-Union (IoU) is a standard metric used to evaluate the accuracy of image segmentation, which calculates the ratio of the intersection and union between the true value and the predicted segmentation. For each type of object, we respectively calculate the results of true positives (TP), false positive (FP), and false negative (FN), and then acquire the IoU of each object using the following formula:

$$\text{IoU} = \frac{TP}{TP + FP + FN} \tag{6}$$

The experimental results of our dataset and YCB Benchmarks are shown in Tables 1 and 2, respectively. Since the mask R-CNN method is not sensitive to image boundaries, the geometric method can clearly discriminate image boundaries, so we combine the deep learning and geometric feature methods to merge and segment. Since it can compensate for the edge and internal defects of mask R-CNN, our method is more accurate than the Mask R-CNN method. The MIoU (Mean Intersection over Union) increased by 2.18% and 5.70% on our dataset and YCB Benchmarks, respectively. Whether the object is large or small, square or round, our method performed better than the Mask R-CNN method in all results, which proved that our algorithm can be suitable for a variety of objects. In extreme cases, like lemon and orange, our method did not perform as good as usual. This is mainly caused by the bad quality of the point cloud. The precision of our method is influenced by the quality of point clouds. When an RGBD camera shoots spherical object, the point clouds of the edges are distorted, which has great effects on the image segmentation. Even so, our method is still more accurate than the Mask R-CNN method. Thus, the applicability and accuracy of our method is better than the Mask R-CNN method. The performance of the two above methods on the two datasets is quite different, because the background on the YCB Benchmarks is not exactly the same as our background, and each image in the datasets contains only one object, while our captured image contains several objects. The Mask R-CNN method preformed quite good in our dataset, it is difficult to improve the precision of segmentation. However, while the Mask R-CNN method preformed poor in the YCB dataset, our method made a greater improvement.

Table 1. Our dataset results on instance segmentation Intersection-over-Union (IoU) (%).

Method	Mean	Cylinder	Half-Cylinder	L-Block	Cuboid	Box
Mask R-CNN [40]	90.782	92.128	90.168	92.788	86.498	92.327
Our method	92.958	92.174	91.330	92.991	92.443	95.852

Table 2. YCB dataset results on instance segmentation IoU (%).

Method	Mean	Lemon	Pear	Orange	Plum	Bowl	Mug
Mask-RCNN [40]	85.221	84.364	82.427	85.868	81.811	87.500	89.356
Our method	90.919	88.194	88.782	91.486	91.440	92.525	93.085

4.3. Three-Dimensional Reconstruction Results

In order to prove the accuracy of the algorithm, we tested the following four methods:

1. Mask-only: mask R-CNN for image segmentation and the forward kinematics for camera position calculation;
2. Mask+ICP: mask R-CNN for image segmentation, the forward kinematics, and ICP registration for camera position calculation;
3. SLIC+ICP: Simple Linear Iterative Cluster (SLIC), the forward kinematics, and ICP registration for camera position calculation; and
4. Our method: mask R-CNN is combined with the LCCCP method for image segmentation, the forward kinematics, and SICP registration for camera position calculation.

After building the 3D model of five types of objects with these methods, we made the ICP match with the ground that was true of the object to calculate the object reconstruction accuracy and the Cloud to Cloud (C2C) absolute distance. The results are shown in Figure 4. Figure 4a shows the original image, and Figure 4b–e represent the 3D reconstruction results of mask-only, mask + ICP, SLIC + ICP, and our method, respectively, with different color points representing different types of objects. Table 3 shows the Cloud to Cloud (C2C) absolute distance between object models and 3D reconstruction by four methods. The higher value of C2C absolute distance means the lower precision of the 3D reconstruction. The comparison results show that as the camera position is inaccurate due to

the robotic manipulator motion error, the mask-only method has the lowest modeling accuracy, and the image of each frame does not overlap well. Since the mask + ICP and the SLIC + ICP method optimizes the camera position, the image coincides well, and the model accuracy is greatly improved compared to the non-optimization method. Our method improves the accuracy of 3D reconstruction based on the mask + ICP method because it improves the segmentation quality of each frame of image.

(**a**) (**b**) (**c**) (**d**) (**e**)

Figure 4. Comparison of the semantic map with four methods. (**a**) The original image, (**b**) the reconstruction results of the mask-only method, (**c**) the reconstruction results of the mask + ICP method, (**d**) the reconstruction results of the SLIC + ICP method, and (**e**) the reconstruction results of our method.

Table 3. Cloud to Cloud (C2C) absolute distances between our dataset models and 3D reconstruction (mm).

Method	Cylinder	Half-Cylinder	L-Block	Cuboid	Box
Mask-only [40]	4.786	5.851	5.622	4.806	3.442
Mask + ICP [34]	3.534	4.597	4.535	3.250	3.083
SLIC + ICP [44]	3.504	4.250	4.380	3.262	3.012
Our method	3.449	4.074	4.142	2.992	2.973

We evaluated the accuracy of 3D reconstruction by the method introduced in [43]. The 3D reconstruction accuracy of the four methods on our dataset is shown in Figure 5. Due to the poor quality of the image segmentation boundary of mask R-CNN, the reconstructed model has a large number of misidentification points. The motion error of the robotic manipulator and the camera calibration error result in an inaccurate position of the camera in the world coordinate system, so, when directly using the mask R-CNN method, the 3D modeling accuracy is low, with only 28.18% of the points in 1 mm distance to the model. Since the mask + ICP method optimizes the camera position, the 3D modeling accuracy is improved compared to the non-optimization method, but as the quality of image segmentation is still poor, only 48.23% of the points are within 1 mm of the model. Our method employs fusion segmentation to improve the quality of image segmentation and also uses the segmentation SICP method to finely correct the image, so it has the highest modeling accuracy among the four methods, and the average precision reaches 53.16%, which is improved by 25.49% over the mask-only method, 4.93% over the mask + ICP method, and 3.50% over the SLIC+ICP method.

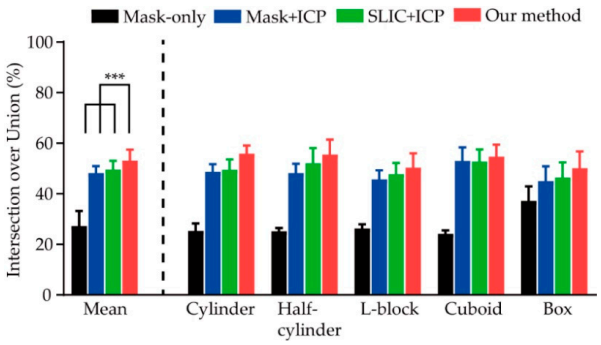


Figure 5. Our dataset results on 3D reconstruction IoU (%). Our method significantly improved the 3D reconstruction IoU compared with the mask-only method, the mask + ICP method, and the SLIC + ICP method by two-way Analysis of Variance (ANOVA) repeat measures with Tukey's multiple comparison test (*** $p < 0.001$).

Similarly, we validate our algorithm on the YCB Benchmarks. The C2C absolute distances between YCB models and 3D reconstruction by four methods are shown in Table 4. The C2C absolute distance can be used to evaluate the similarity between 3D reconstruction results and object models. The lower the value, the higher the accuracy of 3D reconstruction and the more significant the similarity is between the object models. The results in Table 4 indicate that the C2C absolute distances of each object decrease successively in the four methods of mask-only, mask + ICP, SLIC + ICP, and our method, which suggests that the 3D reconstruction results of our method are closer to the object models with highest accuracy. Compared with the other three methods, our method improves the accuracy of image segmentation and reduces the number of outlier points, so the 3D reconstruction results of our method is more accurate. The YCB dataset results on 3D reconstruction are shown in Figure 6. Since the point cloud model of the YCB Benchmarks is obtained by the depth camera in multi-angle shooting, it is closer to the actual situation than the point cloud model generated by Computer Aided Design (CAD), so the four methods perform better on the YCB Benchmarks. As shown in Figure 6, the average accuracy of our method on the YCB Benchmarks is 13.65% over the mask-only method, 4.01% over the mask + ICP method, and 3.27% over the SLIC + ICP method.

Table 4. C2C absolute distances between YCB models and 3D reconstruction (mm).

Method	Lemon	Pear	Orange	Plum	Bowl	Mug
Mask-only [40]	3.727	8.717	4.802	4.918	4.079	3.601
Mask + ICP [34]	3.646	6.244	4.240	4.574	3.341	3.131
SLIC + ICP [44]	3.121	5.849	4.106	4.421	3.251	3.110
Our method	2.593	5.406	3.586	3.693	3.067	2.968

We counted the average CPU time of each methods, as shown in Table 5. Due to the missing geometric feature segmentation and camera pose optimization, the mask-only method ran fastest with lowest precision. Without geometric feature segmentation, the mask + ICP method saved time in segmentation, but the precision of the 3D reconstruction was still poor. The SLIC + ICP method balanced performance and CPU time. Our method took a little more time in segmentation than the SLIC + ICP method, but we saved much more time in the 3D mapping. Because we utilized the SICP method to remove unrelated objects and accelerate point clouds matching.

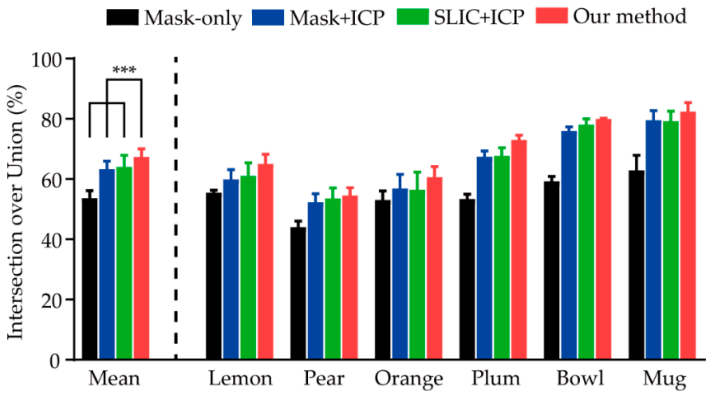


Figure 6. YCB dataset results on 3D reconstruction IoU (%). Our method significantly improved the 3D reconstruction IoU compared with the mask-only method, the mask + ICP method, and the SLIC + ICP method by two-way ANOVA repeat measures with Tukey's multiple comparison test (*** $p < 0.001$).

Table 5. Average CPU Time of each method (ms).

Step	Mask-Only	Mask + ICP	SLIC + ICP	Our Method
Segmentation	81	81	784	802
3D Mapping	12	330	334	271

5. Discussion and Conclusions

This paper proposes an algorithm framework for semantic 3D reconstruction using a robotic manipulator with an eye-in-hand camera. Unlike SLAM, SFM, and other multi-angle modeling methods, our approach adds semantic information into the 3D reconstruction process. We have improved the precision of image segmentation by combining deep learning and geometric feature analysis, and we have increased the accuracy of the 3D reconstruction model through the SICP algorithm to optimize camera pose. The semantic information plays two important roles in 3D reconstruction, one of which is providing the foundation for voxel block merging in image segmentation works, and the other is to remove background during the point cloud matching process and improve the accuracy of the ICP algorithm.

We evaluated the four methods on the YCB Benchmarks and the dataset created by ourselves. The experimental results show that, compared with the deep learning methods, our algorithm is more accurate in the edge segmentation of objects, leading to an improvement of 3D reconstruction. Moreover, the accuracy of the 3D reconstruction of objects is remarkably improved due to the removal of the background interference. Compared with the mask-only, mask + ICP, and SLIC + ICP methods, our method improved the accuracy of the 3D reconstruction on the YCB Benchmarks by 13.65%, 4.01%, and 3.27%, respectively. The same trend was showed on our dataset, with the increasing of the accuracy by 25.49%, 4.93%, and 3.50%, respectively.

In the future, we will apply this method to more scenarios and objects. Based on semantic 3D reconstruction, we will use the object point cloud model to analyze the spatial topological relationship between objects to obtain the decision of the corresponding capture strategy and then make the autonomous robot planning perform a variety of tasks in a semantic map.

Author Contributions: Conceptualization, F.Z. and Y.F. methodology, F.Z. software, Y.F. validation, Y.F., P.W. and W.G. formal analysis, Y.F. writing-original draft preparation, F.Z. writing-review and editing, Y.F.; visualization, Y.F.; supervision, M.L. and H.C. project administration, M.L. funding acquisition, M.L. and X.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by [Natural Science Foundation of China] grant number [61773139], [Shenzhen Science and Technology Program] grant number [KQTD20161125151346 54], and [Shenzhen Special Fund for Future Industrial Development] grant number [JCYJ20160425150757025].

Acknowledgments: This work was supported by Natural Science Foundation of China (No.61773139), Shenzhen Science and Technology Program (No.KQTD20161125151346 54) and Shenzhen Special Fund for Future Industrial Development (No.JCYJ20160425150757025).

Conflicts of Interest: The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

References

1. Foumani, M.; Razeghi, A.; Smith-Miles, K. Stochastic optimization of two-Machine flow shop robotic cells with controllable inspection times: From theory toward practice. *Robot. Comput. Integr. Manuf.* **2020**, *61*, 101822. [CrossRef]
2. Foumani, M.; Smith-Miles, K.; Gunawan, I. Scheduling of two-Machine robotic rework cells: In-Process, post-Process and in-Line inspection scenarios. *Robot. Auton. Syst.* **2017**, *91*, 210–225. [CrossRef]
3. Foumani, M.; Smith-Miles, K.; Gunawan, I.; Moeini, A. A framework for stochastic scheduling of two-Machine robotic rework cells with in-Process inspection system. *Comput. Ind. Eng.* **2017**, *112*, 492–502. [CrossRef]
4. Jia, Z.; Gallagher, A.; Saxena, A.; Chen, T. 3d-Based reasoning with blocks, support, and stability. In Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Portland, OR, USA, 23–28 June 2013; pp. 1–8.
5. Jia, Z.; Gallagher, A.C.; Saxena, A.; Chen, T. 3d reasoning from blocks to stability. *IEEE Trans. Pattern Anal. Mach. Intell.* **2014**, *37*, 905–918. [CrossRef] [PubMed]
6. Zheng, B.; Zhao, Y.; Yu, J.; Ikeuchi, K.; Zhu, S.-C. Scene understanding by reasoning stability and safety. *Int. J. Comput. Vis.* **2015**, *112*, 221–238. [CrossRef]

7. Tremblay, J.; To, T.; Molchanov, A.; Tyree, S.; Kautz, J.; Birchfield, S. Synthetically trained neural networks for learning human-Readable plans from real-World demonstrations. In Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, QLD, Australia, 21–25 May 2018; pp. 1–5.
8. Blodow, N.; Goron, L.C.; Marton, Z.-C.; Pangercic, D.; Rühr, T.; Tenorth, M.; Beetz, M. Autonomous semantic mapping for robots performing everyday manipulation tasks in kitchen environments. In Proceedings of the 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems, San Francisco, CA, USA, 25–30 September 2011; pp. 4263–4270.
9. Galindo, C.; Fernández-Madrigal, J.-A.; González, J.; Saffiotti, A. Robot task planning using semantic maps. *Robot. Auton. Syst.* **2008**, *56*, 955–966. [CrossRef]
10. Silberman, N.; Hoiem, D.; Kohli, P.; Fergus, R. Indoor segmentation and support inference from rgbd images. In Proceedings of the 2012 European Conference on Computer Vision (ECCV), Florence, Italy, 7–13 October 2012; pp. 746–760.
11. Zhang, J.; Singh, S. Visual-Lidar odometry and mapping: Low-Drift, robust, and fast. In Proceedings of the 2015 IEEE International Conference on Robotics and Automation (ICRA), Seattle, WA, USA, 26–30 May 2015; pp. 2174–2181.
12. Paxton, C.; Barnoy, Y.; Katyal, K.; Arora, R.; Hager, G.D. Visual robot task planning. In Proceedings of the 2019 International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 20–24 May 2019; pp. 8832–8838.
13. Endres, F.; Hess, J.; Sturm, J.; Cremers, D.; Burgard, W. 3-D mapping with an RGB-D camera. *IEEE Trans. Robot.* **2013**, *30*, 177–187. [CrossRef]
14. Smith, W.A.; Ramamoorthi, R.; Tozza, S. Height-From-Polarisation with unknown lighting or albedo. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *41*, 2875–2888. [CrossRef] [PubMed]
15. Tozza, S.; Smith, W.A.; Zhu, D.; Ramamoorthi, R.; Hancock, E.R. Linear differential constraints for photo-Polarimetric height estimation. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 2279–2287.
16. Walck, G.; Drouin, M. Progressive 3D reconstruction of unknown objects using one eye-In-Hand camera. In Proceedings of the 2009 IEEE International Conference on Robotics and Biomimetics (ROBIO), Guilin, China, 19–23 December 2009; pp. 971–976.
17. Tozza, S.; Falcone, M. Analysis and approximation of some shape-From-Shading models for non-Lambertian surfaces. *J. Math. Imaging Vis.* **2016**, *55*, 153–178. [CrossRef]
18. Fuchs, S.; May, S. Calibration and registration for precise surface reconstruction with Time-Of-Flight cameras. *Int. J. Intell. Syst. Technol. Appl.* **2008**, *5*, 274–284. [CrossRef]
19. Barth, R.; Hemming, J.; van Henten, E.J. Design of an eye-In-Hand sensing and servo control framework for harvesting robotics in dense vegetation. *Biosyst. Eng.* **2016**, *146*, 71–84. [CrossRef]
20. Chang, W.-C.; Wu, C.-H. Eye-In-Hand vision-Based robotic bin-Picking with active laser projection. *Int. J. Adv. Manuf. Technol.* **2016**, *85*, 2873–2885. [CrossRef]
21. Bescos, B.; Fácil, J.M.; Civera, J.; Neira, J. DynaSLAM: Tracking, mapping, and inpainting in dynamic scenes. *IEEE Robot. Autom. Lett.* **2018**, *3*, 4076–4083. [CrossRef]
22. Brachmann, E.; Rother, C. Learning less is more-6d camera localization via 3d surface regression. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018; pp. 4654–4662.
23. Mur-Artal, R.; Tardós, J.D. Orb-slam2: An open-Source slam system for monocular, stereo, and rgb-d cameras. *IEEE Trans. Robot.* **2017**, *33*, 1255–1262. [CrossRef]
24. Fan, H.; Su, H.; Guibas, L.J. A point set generation network for 3d object reconstruction from a single image. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 605–613.

25. Fehr, M.; Furrer, F.; Dryanovski, I.; Sturm, J.; Gilitschenski, I.; Siegwart, R.; Cadena, C. TSDF-Based change detection for consistent long-Term dense reconstruction and dynamic object discovery. In Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA), Singapore, 29 May–3 June 2017; pp. 5237–5244.
26. Karpathy, A.; Miller, S.; Fei-Fei, L. Object discovery in 3d scenes via shape analysis. In Proceedings of the 2013 IEEE International Conference on Robotics and Automation (ICRA), Karlsruhe, Germany, 6–10 May 2013; pp. 2088–2095.
27. Koppula, H.S.; Anand, A.; Joachims, T.; Saxena, A. Semantic labeling of 3d point clouds for indoor scenes. In Proceedings of the 2011 Advances in Neural Information Processing Systems (NIPS), Granada, Spain, 12–14 December 2011; pp. 244–252.
28. Salas-Moreno, R.F.; Newcombe, R.A.; Strasdat, H.; Kelly, P.H.; Davison, A.J. Slam++: Simultaneous localisation and mapping at the level of objects. In Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Portland, OR, USA, 23–28 June 2013; pp. 1352–1359.
29. Tateno, K.; Tombari, F.; Navab, N. Real-Time and scalable incremental segmentation on dense slam. In Proceedings of the 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Hamburg, Germany, 28 September–2 October 2015; pp. 4465–4472.
30. Sünderhauf, N.; Pham, T.T.; Latif, Y.; Milford, M.; Reid, I. Meaningful maps with object-Oriented semantic mapping. In Proceedings of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Canada, 24–28 September 2017; pp. 5079–5085.
31. Hermans, A.; Floros, G.; Leibe, B. Dense 3d semantic mapping of indoor scenes from rgb-d images. In Proceedings of the 2014 IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, China, 31 May–7 June 2014; pp. 2631–2638.
32. McCormac, J.; Handa, A.; Davison, A.; Leutenegger, S. Semanticfusion: Dense 3d semantic mapping with convolutional neural networks. In Proceedings of the 2017 IEEE International Conference on Robotics and automation (ICRA), Singapore, 29 May–3 June 2017; pp. 4628–4635.
33. McCormac, J.; Clark, R.; Bloesch, M.; Davison, A.; Leutenegger, S. Fusion++: Volumetric object-level slam. In Proceedings of the 2018 International Conference on 3D Vision (3DV), Verona, Italy, 5–8 September 2018; pp. 32–41.
34. Bowman, S.L.; Atanasov, N.; Daniilidis, K.; Pappas, G.J. Probabilistic data association for semantic slam. In Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA), Singapore, 29 May–3 June 2017; pp. 1722–1729.
35. Girshick, R. Fast r-cnn. In Proceedings of the 2015 IEEE international conference on computer vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
36. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, OH, USA, 23–28 June 2014; pp. 580–587.
37. He, K.; Zhang, X.; Ren, S.; Sun, J. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *37*, 1904–1916. [CrossRef] [PubMed]
38. Li, Y.; Qi, H.; Dai, J.; Ji, X.; Wei, Y. Fully convolutional instance-aware semantic segmentation. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 2359–2367.
39. Lin, G.; Milan, A.; Shen, C.; Reid, I. Refinenet: Multi-Path refinement networks for high-Resolution semantic segmentation. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 1925–1934.
40. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask r-cnn. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 2961–2969.
41. Christoph Stein, S.; Schoeler, M.; Papon, J.; Worgotter, F. Object partitioning using local convexity. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, OH, USA, 23–28 June 2014; pp. 304–311.
42. Calli, B.; Walsman, A.; Singh, A.; Srinivasa, S.; Abbeel, P.; Dollar, A.M. Benchmarking in manipulation research: Using the Yale-CMU-Berkeley object and model set. *IEEE Robot. Autom. Mag.* **2015**, *22*, 36–52. [CrossRef]

43. Garcia-Garcia, A.; Orts-Escolano, S.; Oprea, S.; Villena-Martinez, V.; Garcia-Rodriguez, J. A review on deep learning techniques applied to semantic segmentation. *arXiv* **2017**, arXiv:1704.06857.
44. Runz, M.; Buffier, M.; Agapito, L. Maskfusion: Real-Time recognition, tracking and reconstruction of multiple moving objects. In Proceedings of the 2018 IEEE International Symposium on Mixed and Augmented Reality (ISMAR), Munich, Germany, 16–20 October 2018; pp. 10–20.