S & M 1917

Accurate Rapid Grasping of Small Industrial Parts from Charging Tray in Clutter Scenes

Jianmei Wang,[1] Hang Yin,[1] Shaoming Zhang,[1,2*] Popo Gui,[2] and Kunyuan Xu[1]

[1]College of Surveying and Geo-informatics, Tongji University,
1239 Siping Road, Shanghai 200092, China
[2]Institute of Intelligent Vehicle, Clean Energy Automotive Engineering Center, Tongji University,
4800 Caoan Highway, Shanghai 201804, China

(Received January 20, 2019; accepted May 20, 2019)

Keywords: robotic grasping, pose estimation, object segmentation, object detection, red-green-blue and depth (RGB-D) multimodal data

The rapid detection and fine pose estimation of textureless objects in red-green-blue and depth (RGB-D) images are challenging tasks, especially for small dark industrial parts on the production line in clutter scenes. In this paper, a novel practical method based on an RGB-D sensor, which includes 3D object segmentation and 6D pose estimation, is proposed. At the 3D object segmentation stage, 3D virtual and detected bounding boxes are combined to segment 3D scene point clouds. The 3D virtual bounding boxes are determined from prior information on the parts and charging tray, and the 3D detected bounding boxes are obtained from the 2D detected bounding boxes in part detection based on a Single Shot MultiBox Detector (SSD) network in an RGB image. At the 6D pose estimation stage, the coarse pose is estimated by fitting the central axis of the part from the observed 3D point clouds accompanied by a lot of noise, and then refined with part model point clouds by using the iterative closest point (ICP) algorithm. The proposed method has been successfully applied to robotic grasping on the industrial production line with a customer-leverldepth camera. The results verified that grasping speed reaches the subsecond level and that grasping accuracy reaches the millimeter level. The stability and robustness of the automation system meet the production requirement.

1. Introduction

Intelligent manufacturing is an important component of a smart city. In view of the proposal of “made in China 2025”, a continuously increasing number of small and medium-sized enterprises are introducing industrial robots to upgrade their technology. Industrial robots have numerous advantages such as high efficiency, stability, reliability, and good repeatability and adaptability to operate in a high-risk environment. The conventional grasping mode of industrial robots is usually the teaching programming mode, which has disadvantages of low

[*]Corresponding author: e-mail: 08053@tongji.edu.cn

https://doi.org/10.18494/SAM.2019.2307

flexibility, high requirement for placement position, and low fault tolerance. In recent years, with the increasing application of machine vision in industrial production, the vision-guided grasping of industrial robots has continually attracted the attention of researchers and industry leaders. The introduction of vision has greatly expanded the application range of robotic grasping and enhanced the adaptability and reliability of the system. This is the trend of industrial automation development.

Visual guidance modes are divided into monocular and binocular visions. Monocular vision estimates the coarse pose of an object by 2D image matching. Binocular vision estimates the fine pose of an object by 3D point cloud registration. Binocular vision can be obtained using a stereo or depth camera. A stereo camera generates point clouds on the basis of the parallax principle and camera imaging model. For a textureless object, stereo matching will fail because the corresponding point could not be found. A depth camera, whether based on the structured light or time of flight, always acquires 3D information on objects in real time. Currently, a depth camera has become a common sensor in a vision-guided robotic grasping system.

Industrial parts are usually textureless, smooth, and uniformly colored objects. It is vital to extract discriminative feature descriptors that can represent the object shape. In the study of Drost *et al.*,(1) the global model description was based on oriented point pair features (PPFs). During training, all possible pairs of 3D points on a model are described and recorded in a hash table. During detection, sampled pairs of 3D points from the scene are described and used to vote for corresponding object pose hypotheses. The most voted pose clusters can then be refined with the iterative closest point (ICP).(2) Choi and Christensen further augmented the PPFs with color information and its application to a voting-based 6D object pose estimation.(3) In the study of Logoglu *et al.*, two spatially enhanced local 3D descriptors (SPAIR and CoSPAIR) were proposed, and they outperform the state-of-the-art descriptors in both category and instance-level recognition tasks.(4) The efficiency and performance of these methods directly depend on the complexity of the 3D scene, which might prevent their real-time applications.

Most of the best performing 3D detectors follow a view-based paradigm, in which each object is represented by hundreds or even thousands of images, called templates, which describe the object from a discrete set of viewing angles. For example, Hinterstoisser *et al.*(5) create 3115 template views over a quantized color gradient and surface normal. Tolerance to misalignments is achieved by comparing the binarized representation with pixels in a small local neighborhood. The pose retrieved from the best matching template is used as a starting point for subsequent refinement with ICP. Kehl *et al.* and Hodaň *et al.* optimized the matching process using a cascade-type and hash-coded voting scheme, improving the accuracy of 6D pose estimation.(6,7) They achieved a sublinear complexity in the number of trained objects by a hashing search. Although there are hundreds and thousands of templates, only a very small, predefined 6D pose space is covered. Placing the object differently, e.g., on its head, would lead to failure if this view had not been specifically included during training. Unfortunately, additional views increase computation time and add to overall ambiguity in the matching stage.

Over the last few years, deep-learning-based methods have shown promising results of object detection and 6D pose estimation. Wohlhart and Lepetit(8) employed convolutional neural networks (CNNs) to generate descriptors of object views that efficiently capture both the object identity and the 6D pose. Simple similarity and dissimilarity constraints between descriptors–defined by Euclidean distance–are employed to train CNNs. This method has been shown to outperform state-of-the-art methods based on the dataset of Hinterstoisser *et al.*(5) Krull *et al.*(9) presented a model for the posterior distribution in 6D pose estimation, which uses a CNN to map rendered and observed images to an energy value. They observed empirically that CNNs do not specialize on the geometry or appearance of specific objects; moreover, CNNs can be used with objects of vastly different shapes and appearances, and in different backgrounds. Mousavian *et al.*(10) extended Single Shot MultiBox Detector (SSD)(11) to include pose estimates for categories, which infers 3D bounding boxes of objects in urban traffic and regresses 3D box corners and an azimuth angle. Kehl *et al.*(12) extend the SSD paradigm to cover the full 6D pose space and train on synthetic model data only. This method has shown that color-based detectors can indeed match and surpass current state-of-the-art methods that leverage red-green-blue and depth (RGB-D) data while being around one order of magnitude faster. However, the methods based on deep neural networks always require powerful hardware platforms to maintain good running. Moreover, the networks of 3D object detection and 6D pose estimation require many more training samples and much more training time.

Although impressive results have been achieved in 3D object detection and 6D pose estimation from RGB-D images over the last decade, most of the existing methods cannot be used to grasp textureless parts on the production line in clutter scenes, especially small dark parts. The robustness of an automation system is the key to industrial use, because it will directly reduce the production efficiency and even cause severe economic losses if grasping fails. To accurately rapidly grasp small dark textureless parts from a charging tray on the production line in clutter scenes, a novel practical method based on an RGB-D sensor is proposed in this paper. Different from the above methods of simultaneous 3D object detection and 6D pose estimation, the proposed method is a two-stage cascaded method including 3D object segmentation and 6D pose estimation. The first contribution of our method is that 3D virtual and detected bounding boxes are combined to improve the 3D part segmentation accuracy. The 3D virtual bounding box is determined from prior information on the parts and charging tray, and the 3D detected bounding box is obtained from the 2D detected bounding box in part detection based on an SSD network in an RGB image. The second contribution of our method is that the coarse pose is estimated by fitting the central axis of the part rather than the part model to improve the accuracy of the pose estimation. The proposed method has been successfully applied to robotic grasping on the industrial production line. The results verified that grasping speed reaches the subsecond level, and that grasping accuracy reaches the millimeter level, and the stability and robustness of the automation system meet the production requirement.

The remaining sections of this paper are organized as follows. 3D object segmentation and 6D pose estimation are detailed in Sects. 2 and 3, respectively. Section 4 describes the practical application and results of the proposed method. Finally, conclusions and future research plans are given in Sect. 5.

2. 3D Object Segmentation

2.1 Motivation

The industrial production line is a structured environment; the robotic arm usually grasps parts from the charging tray at specific locations. Ideally, the parts are all inserted vertically into the holes in the tray, and 3D point clouds of each part can be isolated using a set of 3D virtual bounding boxes, the size of which is determined by the part and the position of which is determined by the hole array in the tray. The positions of the hole array are known after robotic grasping system calibration. In reality, as shown in Fig. 1, some parts are inserted obliquely into the holes, which leads to the observed point clouds of a part not being covered entirely by a 3D virtual bounding box.

CNN-based 2D/3D object detectors have been widely applied to object location and pose estimation, and have achieved state-of-the-art results on multiple challenge datasets. However, although the object detectors based on CNNs have shown promising results in handling object occlusion and background clutter, the detected bounding boxes cannot be accurately localized.(13) As shown in Fig. 1, the 3D detected bounding box excludes some observed point clouds. For a small part with sparse point clouds, a localization error will severely affect the subsequent model fitting and pose estimation. Considering the above two cases, we proposed to combine the 3D virtual and detected bounding boxes to segment 3D scene point clouds.

2.2 2D object detection based on SSD network

Compared with 3D object detectors based on CNNs, 2D object detectors require a much smaller number of training samples and a shorter training time, which are more suitable for low-cost engineering applications. A majority of modern object detectors are based on two-

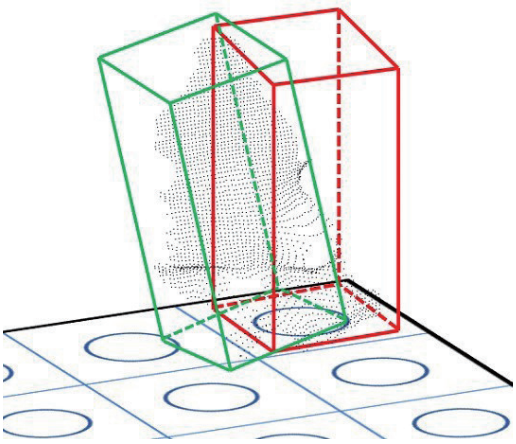

Fig. 1. (Color online) The black point is the observed point clouds of the part, the blue plane is the tray with the hole array, the red bounding box is the 3D virtual bounding box, the size of which is determined by the part and the position of which is determined by the hole array, and the green bounding box is the 3D detected bounding box based on the RGB-D image.

stage frameworks such as R-CNN,(14) SPP,(15) Fast-RCNN,(16) Faster-RCNN,(17) and R-FCN,(18) in which the first phase generates a sparse set of regions of interest (ROIs) and the second phase classifies each proposal by a network. On the other hand, YOLO(19) and SSD(11) have popularized the one-stage approach, which removes the ROI pooling step and detects objects in a single network. One-stage detectors are usually more computationally efficient than two-stage detectors while maintaining a competitive performance. YOLO predicts bounding box coordinates directly from an image and is later improved in YOLO9000(20) by switching to anchor boxes. SSD places anchor boxes densely over feature maps from multiple scales and directly classifies and refines each anchor box. Compared with YOLO, SSD has a significant improvement in mean average precision (mAP).

The architecture of the SSD network is shown in Fig. 2. VGG16(21) is used as the base network and its fully connected layers (FC6, FC7) are converted into convolutional layers. In addition, SSD designs four extra feature layers connected to the end of VGG16. Each extra convolutional layer outputs a feature map, which is used as an input for prediction. These extra layers, together with Conv4_3 and the FC7 layer, predict the offsets to default boxes of various scales and aspect ratios and their associated confidences by small convolutional filters.

2.3 3D object segmentation based on region growing

The output of the SSD network is 2D bounding boxes, which will be back-projected to a 3D space with aligned RGB and depth images as shown in Fig. 3. According to the idea described in Sect. 2.1, the union set of 3D detected and virtual bounding boxes is used to segment 3D scene point clouds. Some noise and the observed point clouds of other parts are included in the initial results. Furthermore, the region growing on the basis of the distance measurement is used to refine the segmentation shown in Fig. 4.

3. 6D Pose Estimation

3.1 Fitting the central axis of the part

From a single view with a depth camera, only partial point clouds of each object can be obtained, and the raw point clouds are more or less distorted and scattered. A higher-level

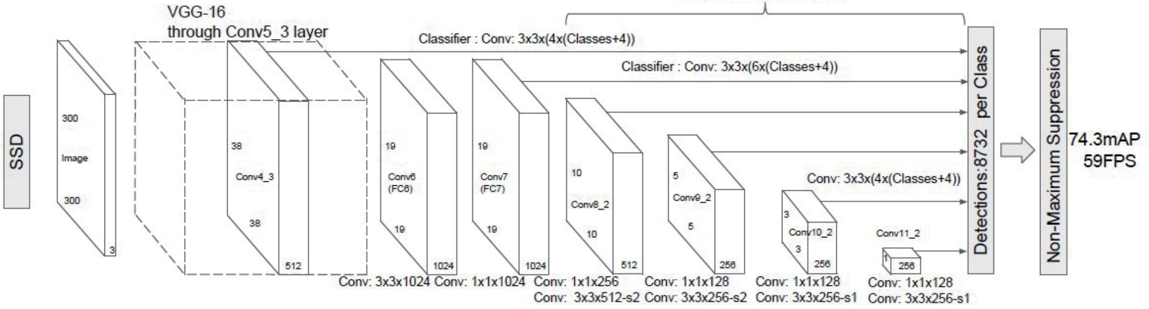

Fig. 2. Architecture of SSD network.(11)

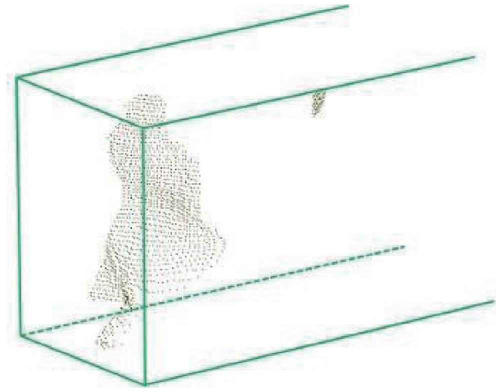

Fig. 3. (Color online) 3D bounding box and 3D isolated point clouds.

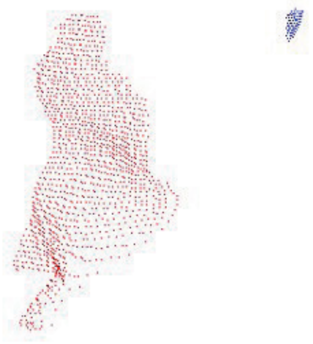

Fig. 4. (Color online) Refined point cloud segmentation based on region growing.

representation of these points as a set of shape primitives (e.g., planes, spheres, or cylinders) obviously gives more valuable clues for grasping. There are a multitude of approaches based on superquadrics for modelling 3D point clouds with shape primitives.(22–24) Superquadrics are a good trade-off between flexibility and computational simplicity, but sensitive to noise and outliers that will cause imperfect approximations.

The diameter of the small part in this paper is about 1 cm. This means that there are only up to 11 point clouds per row reflected by the part when Intel RealSense D415 is installed 1 m away. In contrast, there are more point clouds coming from the tray, as shown in Fig. 5. The part model point clouds are shown in Fig. 5(a), while the observed point clouds are shown in Fig. 5(b). Obviously, it is unreasonable to fit the part model with these 3D point clouds based on superquadrics. Despite a lot of noise, it is not difficult to observe that all the point clouds are roughly symmetrically distributed along the central axis of the part, which inspires us to directly fit the central axis of the part with the point clouds by using the RANSAC algorithm.

3.2 Fine 6D pose estimate

Owing to noise, the fitted posture calculated from the observed point clouds deviates from the true posture of the part model, as shown in Fig. 6. The coarse posture will be refined by the ICP algorithm. ICP is sensitive to the starting value. A good starting value not only increases the convergence speed, but also prevents convergence to the local optimal. As shown in Fig. 7, O'-$X'Y'Z'$ is the starting coordinated system for ICP, which is defined as follows: the Z'-axis is the normal of the fitted plane based on the observed point clouds, the Y'-axis is determined by $Z \times S_1S_1'$, the X'-axis is determined by $Y' \times Z'$, and the origin O' is located at the central point of all point clouds. Theoretically, O'-$X'Y'Z'$ coincides with the O-XYZ coordinated system when ICP finishes; the system is located at the center of the part model and the X-axis is the same as the central axis of the part model. Then, the X-axis becomes the refined posture, and the grasping position shown in Fig. 6 is also obtained.

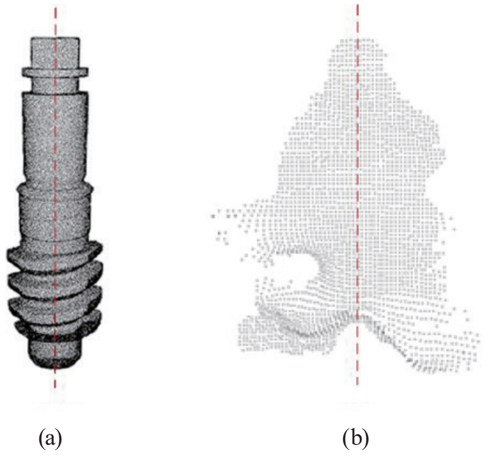

Fig. 5. (Color online) (a) Part model point clouds, and (b) observed point clouds, which contain a lot of noise, but still show rough symmetrical distribution along the central axis of the part.

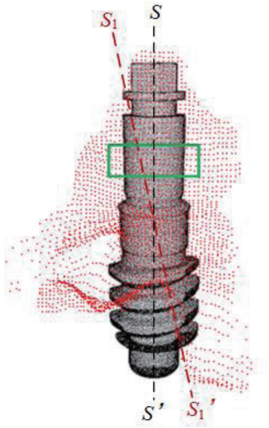

Fig. 6. (Color online) SS' is the true posture of the part mode, S_1S_1' is the fitted posture based on the observed point clouds, and the green rectangle represents the position and opening width of the gripper.

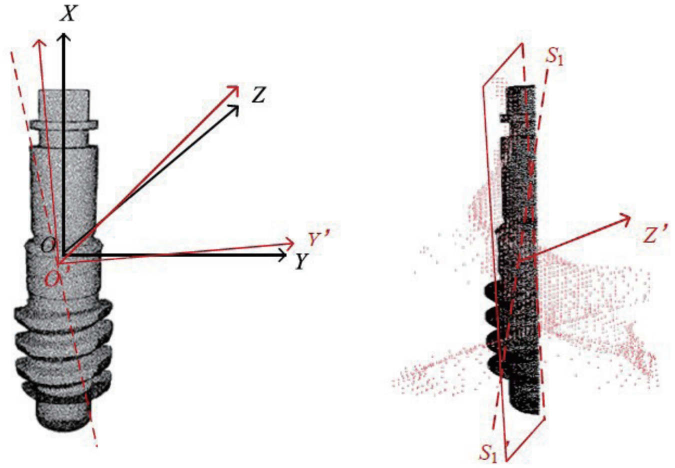

Fig. 7. (Color online) S_1S_1' is the fitted posture based on the observed point clouds and O-XYZ is the coordinated system of the part model. O'-$X'Y'Z'$ is defined as follows: the Z'-axis is the normal of the fitted plane based on the observed point clouds as shown on the right, $Y' = Z' \times S_1 \mathrm{S}_1'$, $X' = Y' \times Z'$, and the origin O' is located at the central point of all point clouds.

4. Experimental Results

4.1 Robotic grasping system

The robotic grasping system and production line are shown in Fig. 8: the orange rectangle shows the YK-XG industrial robotic arm, which provides 4 degrees of freedom, the yellow rectangle shows the gripper, the green rectangle shows Intel RealSensor D415, which is sealed to waterproof and dustproof it, and red arrows denote the heading direction of the production

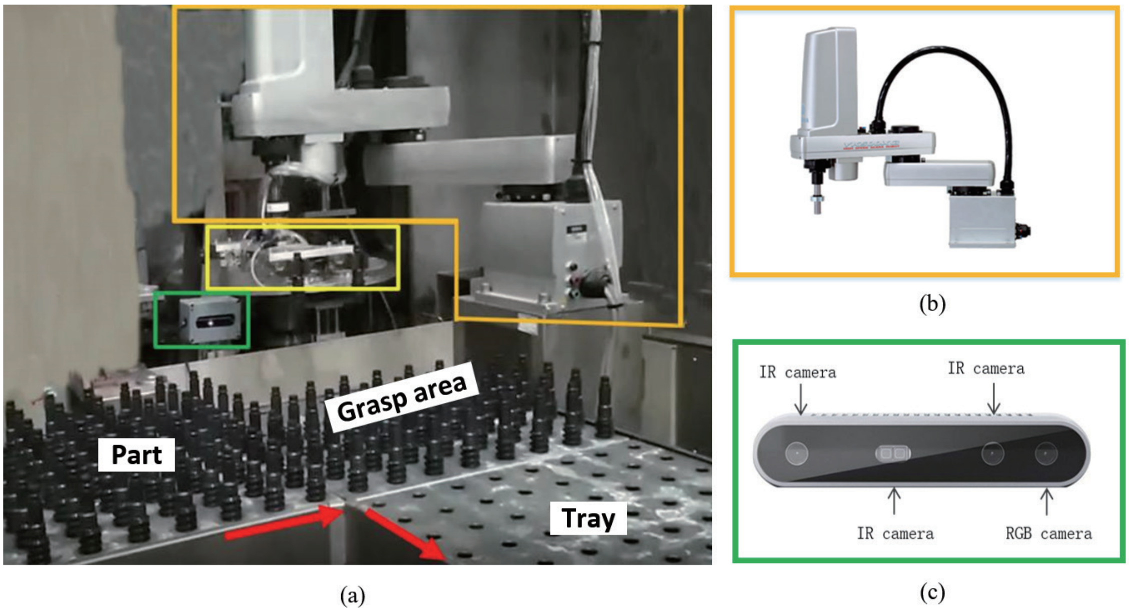

Fig. 8. (Color online) (a) Robotic grasping system and production line. The orange rectangle shows the YK-XG industrial robotic arm, which is shown in detail in (b), the yellow rectangle shows the gripper, the green rectangle shows Intel RealSensor D415, which is shown in detail in (c), and red arrows denote the heading direction of the production line.

line. A tray carries 8 × 8 = 64 parts at a time. In addition to depth camera and hand-to-eye calibrations, the coordinates of the hole array in the tray in the grasping area should be measured before running the system. The automation system is developed in the C++ language in the windows operating system. The CPU frequency of the industrial computer is 3.3 GHz without GPU. The robotic grasping system can operate in the regular working environment of the factory, and no additional illumination source is required.

4.2 3D object segmentation

The training samples are labeled manually using LabelImg software. The training dataset contains 156 images and each image contains not less than 5 parts. Data augmentations, such as rotation and color transform, are applied to ameliorate the diversity of samples. More than 6000 part samples are obtained with data augmentation; which are sufficient for single-target detection. The SSD network is fine-tuned in Caffe in the Ubuntu operating system. The mAP is 90.1%. The detection results based on the SSD network from the RGB image in the production environment, which are satisfactory in the dim and clutter scenes, are shown in Fig. 9. The detected parts are marked with green rectangles with certain probabilities. Clearly, the bounding boxes do not surround the parts exactly.

Object detection based on the SSD network from the RGB image has the risk of omitting the parts. Thus, the 3D detected bounding boxes obtained from the 2D detected bounding boxes in part detection based on the SSD network are combined with the 3D virtual bounding boxes determined by the parts and tray to isolate the scene point clouds shown in Fig. 10; this cannot

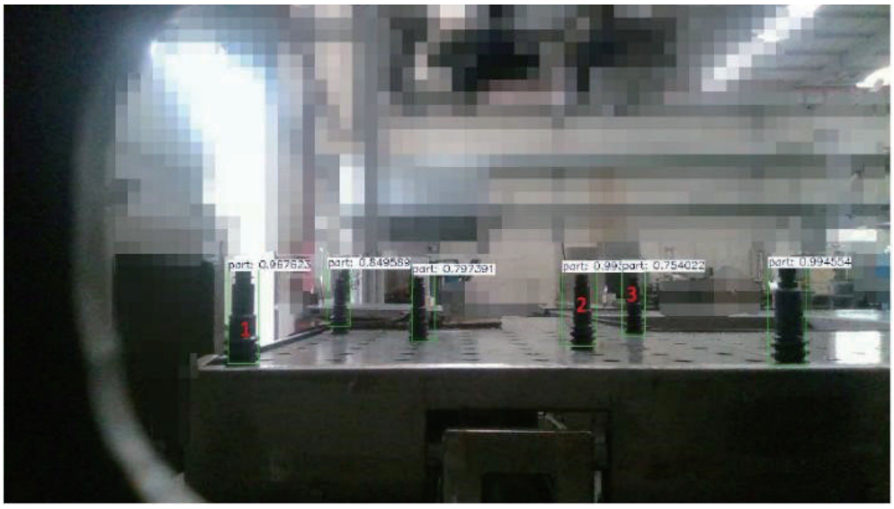

Fig. 9. (Color online) Detection results based on SSD network from RGB image. The detected parts are marked with green rectangles with certain probabilities. To keep the technology secret, the upper half of the image is blurred.

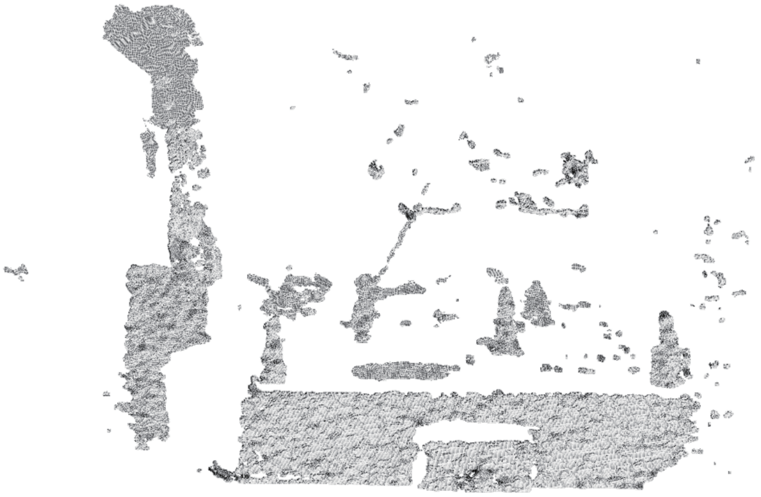

Fig. 10. Scene point clouds.

only ensure that no parts are omitted, but also improve the part segmentation accuracy. From the sizes of the parts and tray, the length and width of the 3D virtual bounding box in this system are both 6 cm. Figure 11 shows the point clouds of the parts labeled with red numbers in Fig. 9.

4.3 6D pose estimation

After 3D point cloud segmentation, the RANSAC algorithm is used to fit the central axis of the part and the plane parallel to the central axis. On the basis of practical experience, the distance thresholds are set to 4 and 2 mm, respectively. The fitted axis is the coarse posture of the part. Next, the part model point clouds are used to refine the coarse posture by using the ICP algorithm. The iteration will stop if any of the following criteria are met: (1) the number of iterations reaches 20; (2) the difference between the current transformation matrix (R,T) and the previous one is smaller than 10^{-6}; (3) the mean squared error (MSE) between the current and previous sets of correspondences is smaller than 0.1 mm; (4) the maximum distance between the current and previous correspondences is smaller than 2 mm. The coarse and refined postures are displayed in red and blue lines in Fig. 12, respectively. The position and posture of grasping the parts in the world coordinated system are detailed in Table 1.

The real scene of robotic grasping is shown in Fig. 13. The time of 3D object segmentation is about 0.20 s, the time of 6D pose estimation is about 0.18 s, and the total time of grasping a part is less than 1 s. The automatic grasping system works continuously for 8 h on the production line without any failure. The opening width of the gripper is 1.7 cm, and the maximum diameter of the part is about 1 cm; this proves that the grasping accuracy is millimeter level. The grasping accuracy decreases with increasing distance, so the robotic grasping system requires that the maximum distance between the depth camera and the parts is limited to 1.5 m based on our experiment.

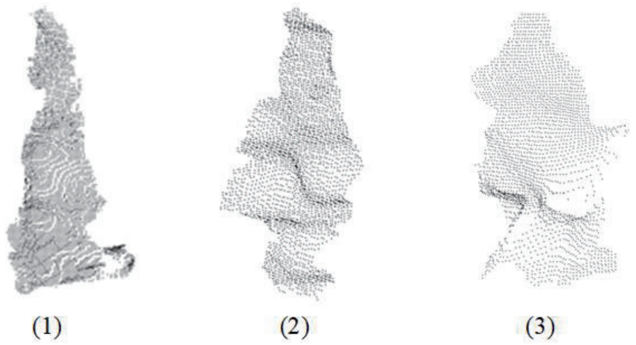

Fig. 11. 3D point cloud segmentation of the detected parts.

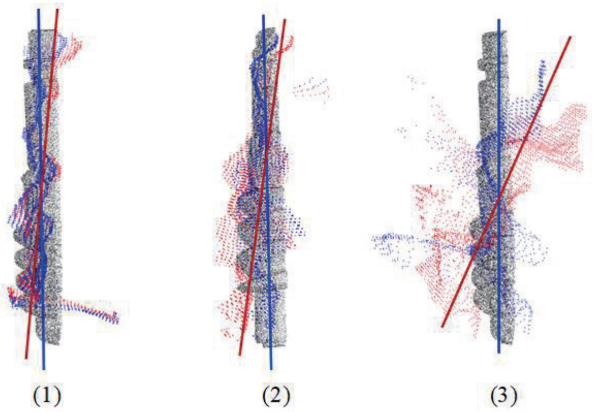

Fig. 12. (Color online) The black points are part model point clouds from the side view. The red point and line are the observed point clouds and the posture fitted by using RANSAC, respectively. The blue point and line are the point clouds observed after translation and the posture refined by using ICP, respectively.

Table 1
Position and posture of grasping the parts in the world coordinated system.

No.	Posture	X (mm)	Y (mm)	Z (mm)	α (rad)	β (rad)	γ (rad)
1	Coarse	3.503	183.550	−494.342	0.067	0.028	1.687
	Refined	−1.188	107.510	−509.576	−0.023	0.164	1.621
2	Coarse	−43.924	147.768	−782.998	0.233	0.024	1.485
	Refined	−17.308	94.595	786.340	0.155	0.047	1.475
3	Coarse	−427.081	98.291	−841.410	0.807	0.124	1.660
	Refined	−480.803	42.463	−812.166	0.810	0.222	1.621

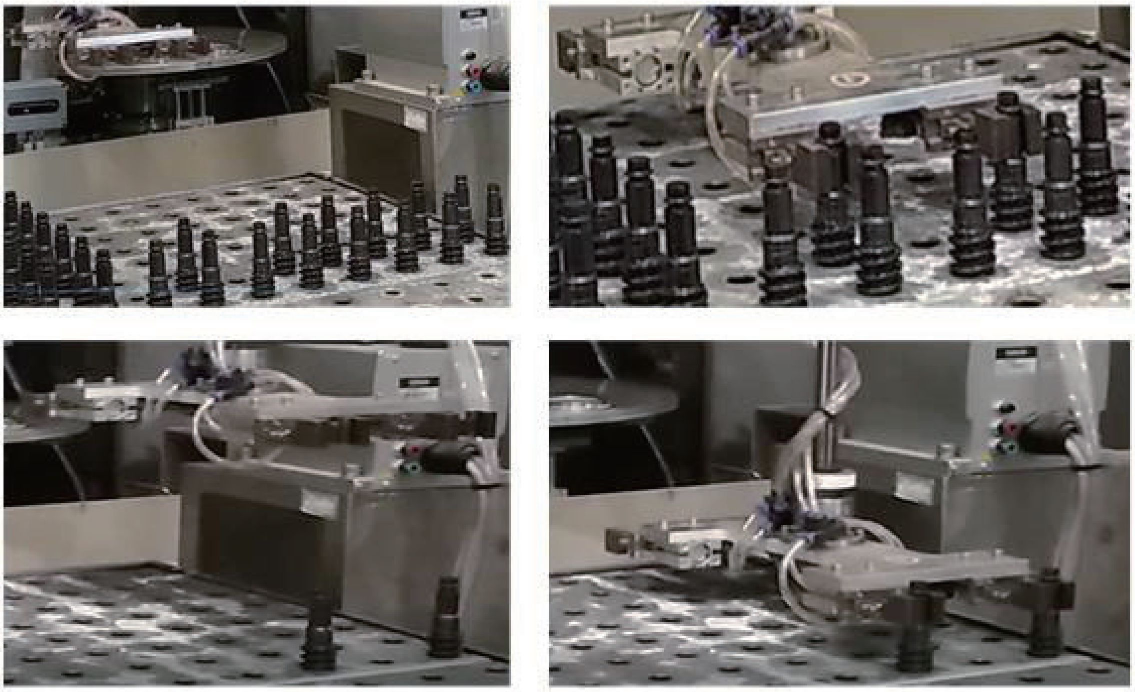
(a) (b)

Fig. 13. (Color online) Real scene of robotic grasping. (a) The scene before grasping and (b) the scene when grasping.

5. Conclusions

In this paper, a practical method of accurately rapidly grasping small dark textureless parts in clutter scenes is proposed. In this method, the position information of the charging tray, combined with object detection results based on the SSD network from RGB images, is used to improve the accuracy of 3D part point cloud segmentation, then the known part model and its symmetry characteristics are used to estimate the 6D pose from the observed point clouds. A millimeter-level grasping accuracy and a subsecond-level grasping speed are achieved using a customer-level depth camera. The practical application on the industrial production line proves the stability and robustness of the proposed method. In this paper, the robotic grasping of regularly placed parts is addressed. In future work, we will focus on bin picking in a heavy clutter scene.

Acknowledgments

This work is sponsored by The National Key Research and Development Program of China (2018YFB0505400) and Science and Technology Commission Shanghai Municipality (16DZ1100701).

References

1 B. Drost, M. Ulrich, N. Navab, and S. Ilic: 2010 IEEE Conf. Computer Vision and Pattern Recognition (CVPR, 2010) 998. https://doi.org/10.1109/CVPR.2010.5540108

2 P. J. Besl and N. D. McKay: Sensor Fusion IV: Control Paradigms Data Struct. **1611** (1992) 586. https://doi.org/10.1109/34.121791

3 C. Choi and H. I. Christensen: IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS, 2012) 3342. https://doi.org/10.1109/IROS.2012.6386067

4 K. B. Logoglu, S. Kalkan, and A. Temizel: Rob. Auton. Sys. **75** (2016) 558. https://doi.org/10.1016/j.robot.2015.09.027

5 S. Hinterstoisser, V. Lepetit, S. Ilic, S. Holzer, G. Bradski, K. Konolige, and N. Navab: Asian Conf. Computer Vision (2012) 548. https://doi.org/10.1007/978-3-642-37331-2_42

6 W. Kehl, F. Tombari, N. Navab, S. Ilic, and V. Lepetit: arXiv preprint arXiv:1607.06062 (2016).

7 T. Hodaň, X. Zabulis, M. Lourakis, Š. Obdržálek, and J. Matas: IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS, 2015) 4421. https://doi.org/10.1109/IROS.2015.7354005

8 P. Wohlhart and V. Lepetit: Proc. IEEE Conf. Computer Vision and Pattern Recognition (2015) 3109. https://doi.org/10.1109/CVPR.2015.7298930

9 A. Krull, E. Brachmann, F. Michel, M. Ying Yang, S. Gumhold, and C. Rother: Proc. IEEE Int. Conf. Computer Vision (2015) 954. https://doi.org/10.1109/ICCV.2015.115

10 A. Mousavian, D. Anguelov, J. Flynn, and J. Košecká: IEEE Conf. Computer Vision and Pattern Recognition (CVPR, 2017) 5632. https://doi.org/10.1109/CVPR.2017.597

11 W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg: Eur. Conf. Computer Vision (2016) 21. https://doi.org/10.1007/978-3-319-46448-0_2

12 W. Kehl, F. Manhardt, F. Tombari, S. Ilic, and N. Navab: Proc. Int. Conf. Computer Vision (ICCV, 2017) 22. https://doi.org/10.1109/ICCV.2017.169

13 B. Jiang, R. Luo, J. Mao, T. Xiao, and Y. Jiang: Eur. Conf. Computer Vision (2018) 816. https://arxiv.org/abs/1807.11590

14 R. Girshick, J. Donahue, T. Darrell, and J. Malik: Proc. IEEE Conf. Computer Vision and Pattern Recognition (2014) 580. https://doi.org/10.1109/CVPR.2014.81

15 K. He, X. Zhang, S. Ren, and J. Sun: Eur. Conf. Computer Vision (2014) 346. https://doi.org/10.1007/978-3-319-10578-9_23

16 R. Girshick: Proc. IEEE Int. Conf. Computer Vision (2015) 1440. https://doi.org/10.1109/ICCV.2015.169

17 S. Ren, K. He, R. Girshick, and J. Sun: Advances in Neural Information Processing Systems (2015) 91. https://doi.org/10.1109/TPAMI.2016.2577031

18 J. Dai, Y. Li, K. He, and J. Sun: Advances in Neural Information Processing Systems (2016) 379.

19 J. Redmon, S. Divvala, R. Girshick, and A. Farhadi: Proc. IEEE Conf. Computer Vision and Pattern Recognition (2016) 779. https://doi.org/10.1109/CVPR.2016.91

20 J. Redmon and A. Farhadi: arXiv preprint (2017). https://doi.org/10.1109/CVPR.2017.690

21 K. Simonyan and A. Zisserman: Comput. Sci. arXiv:1409.1556 (2014) (revised, v6). https://arxiv.org/abs/1409.1556

22 L. Chevalier, F. Jaillet, and A. Baskurt: J. Winter School of Computer Graphics **11** (2003) 1.

23 G. Biegelbauer and M. Vincze: IEEE Int. Conf. Robotics and Automation (2007) 1086. https://doi.org/10.1109/ROBOT.2007.363129

24 R. Pascoal, V. Santos, C. Premebida, and U. Nunes: IEEE Trans. Veh. Technol. **64** (2015) 441. https://doi.org/10.1109/TVT.2014.2321899

About the Authors

Jianmei Wang received her B.S. degree in surveying and mapping engineering and her M.S. degree in cartography and geographic information engineering from Tongji University, Shanghai, China, in 1994 and 1997, respectively, and her Ph.D. degree in photogrammetry and remote sensing from Wuhan University, Wuhan, China, in 2007. Since 1999, she has been a lecturer at Tongji University. Her research interests are in computer vision, spatial data mining, and remote sensing image analysis. (97031@tongji.edu.cn)

Hang Yin receive his B.S. degree in surveying and mapping engineering from Tongji University, Shanghai, China, in 2018. Since 2018, he has been a masteris student at Tongji University. His research interests are in computer vision. (1832023@tongji.edu.cn)

Shaoming Zhang received his B.S. degree in electronics engineering from Tianjin University, Tianjin, China, in 2002, his M.S. degree in communication engineering from the 14th Electronics Institute of Ministry of Information Industry in 2005, and his Ph.D. degree in photogrammetry and remote sensing from Tongji University, China, in 2008. From 2008 to 2014, he was an assistant professor at Tongji University. Since 2014, he has been an associate professor at Tongji University. His research interests are in deep learning, computer vision, and SLAM. (08053@tongji.edu.cn)

Popo Gui received his B.S. degree in surveying and mapping engineering from Wuhan University, Wuhan, China, in 2012, and his M.S. degree in photogrammetry and remote sensing from the Tongji University, Shanghai, China, in 2015. From 2016 to 2017, he has been an algorithm engineer in Baidu. His research interests are in automatic driving.
(ppgui@outlook.com)

Kunyuan Xu received his B.E. degree from Tongji University, Shanghai, China, in 2016. From 2016 to the present, he has been a master student at Tongji University. His research interests are in deep learning and object detection. (kyxu360@hotmail.com)