Proceeding of the IEEE
International Conference on Robotics and Biomimetics
Dali, China, December 2019

Deep-Learning based Robust Edge Detection for Point Pair Feature-based Pose Estimation with Multiple Edge Appearance Models

Diyi Liu, Shogo Arai, Fuyuki Tokuda, Yajun Xu, Jun Kinugawa, Kazuhiro Kosuge
Department of Robotics
Tohoku University
6-6-01 Aoba, Aramaki, Aoba-ku, Sendai 980-8579 Japan
diyi.liu.890101@outlook.com
s_arai@irs.mech.tohoku.ac.jp

***Abstract*— To realize a robotic bin picking system, pose estimation for the objects randomly piled up in a bin is necessary. For various types of objects, many pose estimation algorithms have been proposed so far. Point Pair Feature-based Pose Estimation with Multiple Edge Appearance Models (PPF-MEAM) has been proposed for estimating the pose of industrial parts including some parts whose point clouds are defective in our previous work. Although this method shows high performance in pose estimation under a constant environment, its performance drops under the changing light conditions without tuning parameters. To overcome this problem, we propose Deep-Learning based Robust Edge Detection (DLED) for PPF-MEAM to make it robust to changes of the light. The effectiveness of DLED is proved by the edge detection experiment under different light conditions. Moreover, the pose estimation experiment proves that DLED could improve the pose estimation performance of PPF-MEAM under different light conditions.**

***Index Terms*— Edge Detection, Deep Learning, Pose Estimation, PPF-MEAM, Robotic Bin Picking**

I. Introduction

Automating bin picking task is an urgent need in recent years. In the U.S., 38 percent of the manufacturing labor force moves parts from bins to manufacturing machines. However, 500000 such jobs remain unfilled[1].

To realize the robotic bin picking system, we need to solve the key issue first, the pose estimation of the parts in the bin. Locating a distinct part in a bin, namely, the pose estimation has become an available task with a lot of researches and developments of three-dimensional measurement [2], [3] and introduction of commercialized 3D sensors [4], [5]. 3D point cloud processing has been performed in many fields, such as pose estimation [6], object recognition [7], visual servoing [8], and so on.

Although many researches of pose estimation have been carried out so far, only a few algorithms are used in the realistic production line. One of the challenges is the robustness to the light environment. When the light condition changes, tuning parameters or equipping lighting devices are necessary for some algorithms, which increases the cost of the system. For tunning parameters, experts are needed, which increases the labor cost. For equipping lighting devices, both equipment cost and labor cost are increased. Thus, the robustness to the light environment is important for those factories that have changing light conditions. Besides, the robustness of the pose estimation system could also reduce the time for setting up the light environment.

Therefore, an ideal pose estimation algorithm required by the realistic industry is supposed to be robust to the light conditions. The algorithm should keep high performance when the light condition is changed without tunning any parameters. However, this requirement is rarely discussed in many previous works.

Existing pose estimation algorithms could be categorized according to their input dimensions: 2D, 3D, and 2D plus 3D.

The camera is used to capture the scene data for those algorithms with 2D input. Some algorithms [7], [9], [10] based on 2D information show the high computation speed in pose estimation without discussing the robustness to the light conditions. Although some algorithms [11], [12], [13] shows the robustness to the light conditions, the complex sensor system including a multi-flash camera is necessary, which is not commercialized. Besides, algorithms based on 2D information share one problem. When the target object is far from the camera, the difference in depth only shows a little change in a 2D image. This probably makes such methods lose their precision in recovering depth information. That could be the reason why more algorithms are proposed using 3D measurement sensors.

Many algorithms [14], [15] using 3D information as the input of pose estimation have emerged in recent years. These methods could retain the robustness to light conditions by choosing a 3D sensor that is robust to changes of the light condition. Some 3D sensors based on stereo vision with random point pattern, such as Ensenso N35, could realize this easily.

978-1-7281-6321-5/19//$31.00 ©2019 IEEE

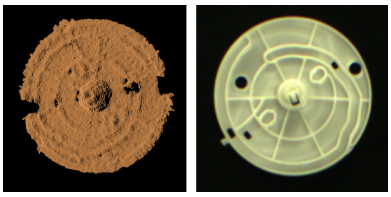


Fig. 1: Defective point cloud of Object 1. The captured point cloud is so insufficient that some existing methods such as [16] could not estimate the horizontal rotation.

However, for some objects whose point cloud is defective captured by the 3D measurement sensor, using 3D data is not enough to estimate the 6D pose of objects. For the part shown in Fig. 1, the point cloud captured by 3D sensor is different from its real appearance. This object is made from the resin with 3 mm thick ridges on its surface. This is too thin to be captured by currently commercialized 3D sensors. The captured 3D data around the plane surface are so insufficient that the horizontal rotation could not be estimated with existing methods [16].

To solve this problem, we have proposed Point Pair Feature-based Pose Estimation with Multiple Edge Appearance Models (PPF-MEAM) [6] that is using both 2D data and 3D data in our previous work. In this research, we have utilized a descriptor named Boundary-to-Boundary-using-Tangent-Line point pair (B2B-TL) computed from the boundary point cloud of the scene. To obtain the boundary point cloud, we perform edge detection on the scene image using Canny Edge detector first. Next, we extract boundary points, points that are corresponding edge pixels in the scene image, from the point cloud of the scene. Using the boundary point cloud, we could estimate the full 6D pose of Object 1. This method shows the high performance in pose estimation in terms of success rate and computation speed.

However, this method heavily relies on the lighting environment. Although some researches based PPF-MEAM has been conducted, such as [17], this problem could not be solved because of the usage of Canny Edge detector in this framework. In the step of edge detection, Canny Edge detector needs to change its parameters for different light conditions. This makes PPF-MEAM not robust to the changes of light conditions without tunning parameters carefully.

In order to solve this problem, we need a robust edge detection method for PPF-MEAM. Before the appearance of neural networks technologies, there are five major categories of edge detection techniques [18]. They are first-order gradient edge detection, second-derivative method, detection using Laplacian of Gaussian (Marr-Hildreth filter), detection using the derivative of Gaussian and the colored edge detector.

For all five types of methods mentioned above, parameters are required to be tuned when the light condition has been changed. Besides, precisely setting a specific threshold for determining an edge is difficult for people who do not have expertise. Recently deep convolutional networks have demonstrated remarkable ability of detecting image edges. There are many methods [19], [20], [21] showing high performance to detect edges. However, a few works have discussed the detecting edges under different light environment.

In this research, we propose Deep-Learning based Robust Edge Detection (DLED) for PPF-MEAM to make it estimate pose of parts robustly regardless of changes of the light condition. The structure of the neural network is originally proposed in [21]. We train this network in a novel way and implement it into PPF-MEAM. We call this pose estimation algorithm DLED PPF-MEAM.

This paper is organized as follows. We first introduce the overview of the DLED PPF-MEAM. Then we introduce the method of edge detection, DLED, and the way for making training data for it. Next, the evaluation experiment of edge detection and pose estimation using four types of different objects are explained. Finally, the conclusion of this research is made.

II. Overview of DLED PPF-MEAM

As shown in Fig. 2, DLED PPF-MEAM differs from the original PPF-MEAM for using DLED but not Canny Edge Algorithm. In the offline phase, a hash table has to be made in advance to describe the appearance of the object, which is the same as the original PPF-MEAM. Besides that, a fully convolutional neural network has to be trained for edge detection.

In the online phase, the pose of the parts in the bin is estimated. We capture the organized point cloud first using a sensor module which includes a color camera sensor and a 3D camera. For the following processing, we obtain the relative transformation from the 3D sensor to the camera by calibrating them in advance. Next, we forward the image of the scene to DLED, a pre-trained fully convolutional networks, to do edge detection and extract the boundary points from the scene point cloud by finding their corresponding 2D edge pixels in the scene image.This process could be easily conducted because we have calibrated the color camera sensor and a 3D camera. Then, the B2B-TL is computed for the boundary points of the scene. After that, we perform the voting scheme proposed by Drost et al. [16] to obtain several pose candidates. Some incorrect pose candidates are removed by performing the pose verification proposed by Li et al. [22]. Instead of using the point cloud of the model, the MEAM is used to perform the pose verification. After this step, several pose candidates could be obtained for each appearance model. These pose candidates are clustered then, and the pose that has the highest pose verification score in each cluster is used to represent that cluster. In the last step, we use the Iterative Closest Point (ICP) algorithm [23] to refine those pose candidates. Refined poses are output and sent to the robot for performing the picking motion.

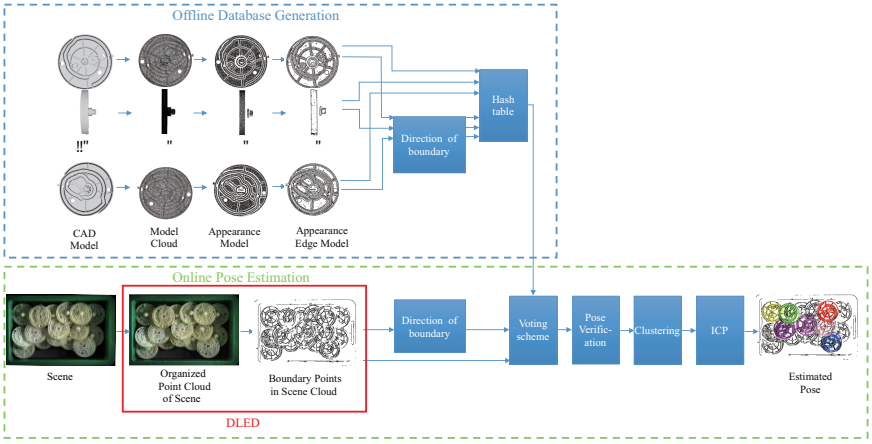


Fig. 2: DLED PPF-MEAM pipeline. This pipeline differs from the original PPF-MEAM for using DLED but not Canny Edge detector as shown in the red box. A database is made to describe the appearance of the object in the offline phase. In the online phase, given the organized point cloud of the scene, we estimate the pose of parts in the bin.

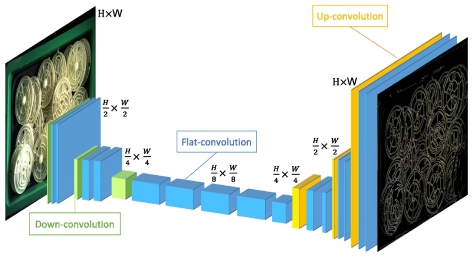


Fig. 3: The network structure of DLED proposed in [21]. Please note that each of the convolutional layer blocks in the figure is proportional to the number of filters it has.

III. Deep-Learning based Robust Edge Detection(DLED)

A. Architecture of Convolutional Networks

As shown in Fig. 3, this model has three types of convolutional layers: down-convolution; flat-convolution and up-convolution. The image size is decreased initially with down-convolution to reduce the data dimension and increase the spatial support of subsequent layers. The image size is restored to its original size using the up-convolution layers. As the loss function, the mean square error criterion

$$E = \frac{1}{n}\sum_{i=1}^{n}(Y_i - Y_i^*)^2, \tag{1}$$

is used, where Y_i and Y_i^* are the ith value of the model output and target output, n is the data points. The learning rate is adapted by using Adadelta.

After we forward the image of the scene to this network, the edge image of the scene could be obtained. To filter out some wrong edges, we set a threshold e_{thre} to binarize the image and perform an algorithm proposed by Hilditch [24] to refine the edge.

B. Training Data

In this research, we use four types of industrial parts. They are Object 1, Object 2, Object 3 , and Object 4. We have taken about 180 pictures of each object. In each scene, one type of objects is piled up randomly in a bin.

For each scene, we took the picture three times under three different light conditions as shown in Fig. 4. For the image taken under the adjusted light, we utilize Canny Edge algorithm to obtain the edge image and use it as the ground truth of this scene. For another two images taken under the darker light condition, we use the same ground truth.

The RGB image of the scene is converted to the gray-scale image first. Then, each pixel is divided by 255. The output image is normalized with the same manner.

We trained our model 50 epochs for each part with a batch size of one. We implement our code under the framework of Keras using TensorFlow as the back end. This takes about 14 hours using two Nvidia GeForce GTX 1080 GPU.

IV. Evaluation Experiment

A. Edge Detection

To evaluate the performance of edge detection, we have compared the detected edge under different light conditions while using Canny Edge and DLED. For each scene, we took

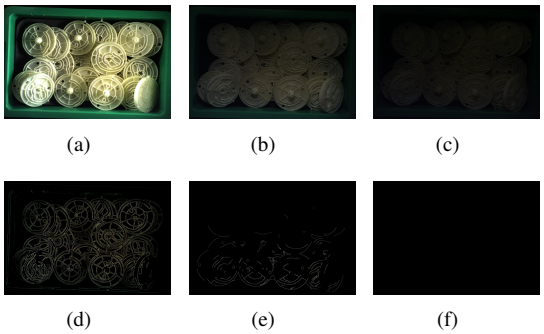


Fig. 4: (a), (b) and (c) are images of the same scene taken under different light conditions, in which (a) is taken under the adjusted light condition. (d), (e) and (f) are edge images obtained by performing Canny Edge Algorithm. We used (d) as the ground truth of edge detection for (a), (b) and (c). We keep the original contrast of the image to emphasize the different result of edge detection.

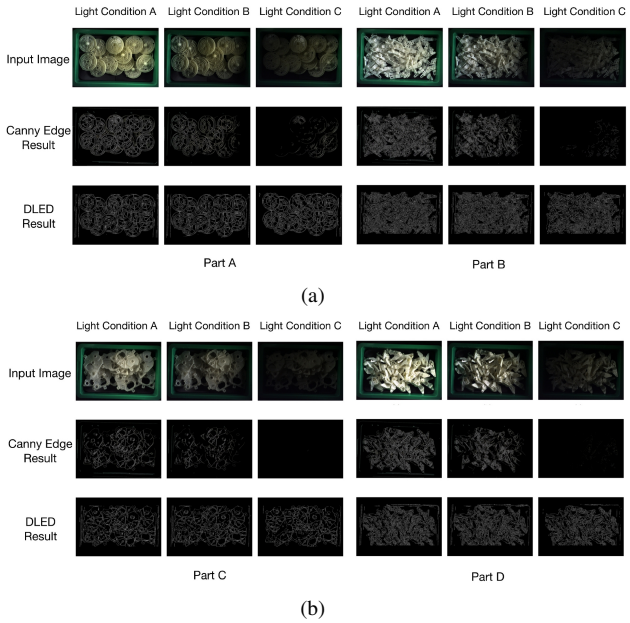


Fig. 5: (a) and (b) are edge detection results shown in original images of these four types of parts. For each scene, three images are taken under different light conditions. Condition A is the adjusted light condition. Condition B is darker than Condition A and Condition C is darker than Condition B. Please note that edge pixels are well detected under very dark light condition compared with using Canny Edge Algorithm. During edge detection, no parameters are tuned.

the picture three times under three different light conditions. During this experiment, we did not change any parameters for these two methods. The result of four different types of parts is shown in the Fig. 5. From these images, we could observe that DLED outperforms Canny Edge under the changing light conditions. As DLED has been trained using images taken under different light conditions, it retains the robustness to changes of the light condition without tuning any parameter for it. According to the result of the validation experiment, we consider the performance of this network is high enough to be used in PPF-MEAM.

B. Pose Estimation

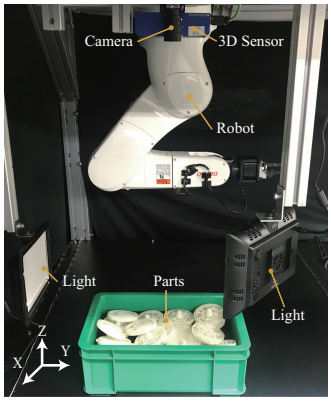
In order to evaluate the performance of the proposed method on the real scene data, an experiment has been conducted and its result is shown in this section. To make it easy to understand the contribution of this work, we compared the performance of methods, PPF-MEAM with Canny Edge and DLED PPF-MEAM.

The program for object detection is implemented in Python while the one for pose estimation is implemented in C++. Those programs are run on one computer which has an Intel Core i7 6950X CPU, 32G RAM and a GTX1080 GPU. To see the best performance of the proposed method in real use, pose estimation programs in this experiment are accelerated by the OpenMP frame. We used Ensenso N35 as the 3D sensor the iDS USB 3 uEye as the 2D camera.

To provide the different lighting environment, two light-emitting diodes (LEDs) are set near the bin for providing a stable lighting environment. This light has a switch knob to adjust the luminance and a screen showing it. Therefore, we could change the luminance to a defined extent. In this experiment, we set three different light conditions. The first condition, Light Condition A, is the well-adjusted condition. Under this condition, the value of luminance of the left light is set to six while the right one is set to 24. Under the second condition, Light Condition B, the value of luminance of the left light is set to six while the right one is set to zero. Under the third condition, Light Condition C, the value of luminance of the left light is set to zero while the right one is set to one.

We tested 20 scenes of four types of parts. They are Object 1, Object 2, Object 3 and Object 4. In each scene, one type of objects is randomly piled up in a bin. We captured the organized point cloud of each scene under three different light conditions. For DLED, e_{thre} is 30. For other parameters mentioned in PPF-MEAM, τ_{d} that makes $d_{\mathrm{dist}} = \tau_{\mathrm{d}} \times D$ was equal to 0.07, where D is the maximum distance between two points on the object. The d_{angle} was set to 3.6° according to our experience. To obtain a good trade-off between recognition rate and computation time. The P_{referred} is set to 20%, 20%, 10%, 20% for Object 1, Object 2, Object 3 and Object 4, respectively. And the P_{ref} is set to 10%, 50%, 10%, 20% for Object 1, Object 2, Object 3 and Object 4. For other parameters we used the same parameters as shown in the paper of PPF-MEAM [6].

We output five pose results in each scene for Object 1,



(a)

Fig. 6: The experimental system was used to verify the DLED PPF-MEAM. We mounted sensor right above the bin. Two light-emitting diodes (LEDs) are installed on both sides of the box for changing the light condition.

Object 2 and Object 4 and three pose results for Object 3. The correctness of pose result is judge manually. we counted the number of true positives and the success rate is the number of true positives over the number of output poses.

The success rate and computation time of the proposed method for every parts are presented in Tables I, II, III and IV respectively. Some examples are shown in Fig 7. We compared DLED PPF-MEAM with PPF-MEAM. Comparing with the PPF-MEAM, the proposed method takes a little bit more time. However, it could estimate the pose of parts with high performance even under the very dark light condition. The proves that DLED PPF-MEAM has more robustness to changes of the light conditions than original PPF-EMAM.

V. Conclusion

In this research, Deep-Learning-based Robust Edge Detection for Pose Estimation (DLED PPF-MEAM), has been proposed. By applying DLED, the edge could be detected successfully even the light condition under very dark without tuning parameters. According to the result of pose estimation experiment, DLED PPF-MEAM shows the high performance of pose estimation regardless of changes of the light condition. This proves that DLED PPF-MEAM could be used in the factories that have changing light conditions. Moreover, using the proposed method simplifies the setup of lighting devices and parameter tuning, which makes the proposed method be easily used in the real factory.

References

[1] E. Truebenbach, “Is fully automated bin picking finally here?” Available at https://www.therobotreport.com/fully-automated-bin-picking-finally-here/, Jan. 2019.

[2] N. Chiba, S. Arai and K. Hashimoto, ”Feedback projection for 3D measurements under complex lighting conditions,” in American Control Conference (ACC), IEEE, 2017, pp. 4649-4656.

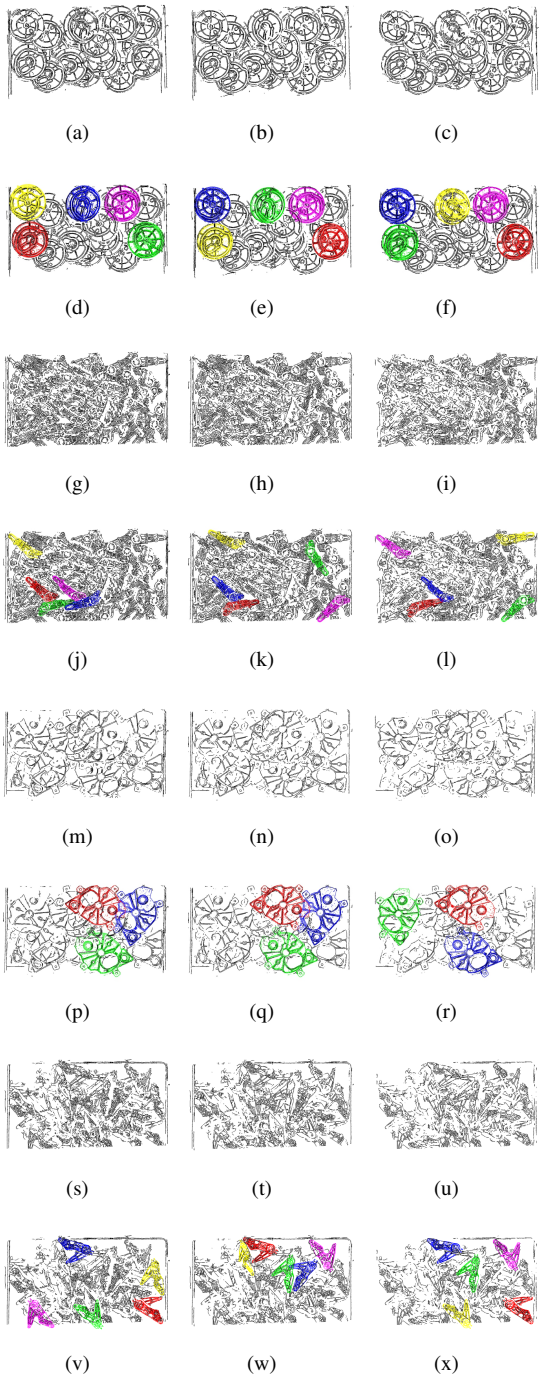


Fig. 7: (a), (b), (c), (g), (h), (i), (m), (n), (o), (s), (t) and (u) are mapped boundary point cloud of the scene under three different light conditions. Corresponding scene images and edge images are shown in Fig. 5. (d), (e), (f), (j), (k), (l), (p), (q), (r), (v) , (w) and (x) are the pose estimation result. To show the result of pose, we transformed the point cloud of the model into the scene space using pose results and rendered them with different colors. These colors are used to show the recommendation rank for graping. The order of recommendation is red, green, blue, yellow and mega, respectively.

TABLE I: Success rate and computation time of pose estimation for real scenes of Object 1.

	Light Condition A		Light Condition B		Light Condition C	
	Rate	Time [ms]	Rate	Time [ms]	Rate	Time [ms]
PPF-MEAM	95.0%	1575	90.0%	1326	17.0%	499
DLED PPF-MEAM	96.0%	1770	95.0%	1735	90.0%	1699

TABLE II: Success rate and computation time of pose estimation for real scenes of Object 2.

	Light Condition A		Light Condition B		Light Condition C	
	Rate	Time [ms]	Rate	Time [ms]	Rate	Time [ms]
PPF-MEAM	96.0%	1694	96.0%	1305	6.0%	325
DLED PPF-MEAM	96.0%	1917	96.0%	1908	93.0%	1767

TABLE III: Success rate and computation time of pose estimation for real scenes of Object 3.

	Light Condition A		Light Condition B		Light Condition C	
	Rate	Time [ms]	Rate	Time [ms]	Rate	Time [ms]
PPF-MEAM	96.7%	1117	85.0%	854	0.0%	128
DLED PPF-MEAM	96.7%	1058	96.7%	1490	90.0%	1331

TABLE IV: Success rate and computation time of pose estimation for real scenes of Object 4.

	Light Condition A		Light Condition B		Light Condition C	
	Rate	Time [ms]	Rate	Time [ms]	Rate	Time [ms]
PPF-MEAM	96.0%	1058	85.0%	836	10.0%	284
DLED PPF-MEAM	97.0%	1319	95.0%	1306	91.0%	1251

[3] M.Y. Liu, , O. Tuzel, A. Veeraraghavan, R. Chellappa, A.Agrawal, H.Okuda, "Pose estimation in heavy clutter using a multi-flash camera," in Robotics and Automation (ICRA), IEEE International Conference on, IEEE, 2010, pp. 2028-2035

[4] Ensenso, "Stereo 3d Cameras" Available at https://en.ids-imaging.com/ensenso-stereo-3d-camera.html, Jan. 2019.

[5] Solomon Technology Corporation, "Vision with Intelligence" Available at https://www.solomon-3d.com/, Jan. 2019.

[6] D. Liu, S. Arai, J. Miao, J. Kinugawa, Z. Wang, and K. Kosuge. "Point pair feature-based pose estimation with multiple edge appearance models (PPF-MEAM) for robotic bin picking," in Sensors, vol. 18, no. 8, pp. 2719, 2018.

[7] A. Collet, D. Berenson, S. S. Srinivasa, and D. Ferguson, Object recognition and full pose registration from a single image for robotic manipulation, in Robotics and Automation (ICRA), IEEE International Conference on. IEEE, 2009, pp. 48-55.

[8] C. Kingkan, S. Ito, S. Arai,T. Nammoto, K. Hashimoto. "Model-based virtual visual servoing with point cloud data," in Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on. IEEE, 2016, pp. 5549-5555

[9] A. Mousavian, D. Anguelov, J. Flynn, and J. Kosecka , 3d bounding box estimation using deep learning and geometry, in Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on. IEEE, 2017, pp. 5632-5640.

[10] P. Poirson, P. Ammirato, C. Y. Fu, W. Liu, J. Kosecka, and A. C. Berg, Fast single shot detection and pose estimation, in 3D Vision (3DV), 2016 Fourth International Conference on. IEEE, 2016, pp. 676-684.

[11] J. J. Rodrigues, J.-S. Kim, M. Furukawa, J. Xavier, P. Aguiar, and T. Kanade, 6d pose estimation of textureless shiny objects using random ferns for bin-picking, in Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on. IEEE, 2012, pp. 3334-3341.

[12] N. Shroff, Y. Taguchi, O. Tuzel, A. Veeraraghavan, S. Ramalingam, and H. Okuda, Finding a needle in a specular haystack, in Robotics and Automation (ICRA), 2011 IEEE International Conference on. IEEE, 2011, pp. 5963-5970.

[13] M. Y. Liu, O. Tuzel, A. Veeraraghavan, Y. Taguchi, T. K. Marks, and R. Chellappa, Fast object localization and pose estimation in heavy clutter for robotic bin picking, in International Journal of Robotics Research, vol. 31, no. 8, pp. 951-973, 2012.

[14] R. B. Rusu, N. Blodow, Z. C. Marton, and M. Beetz, Aligning point cloud views using persistent feature histograms, in Intelligent Robots and Systems (IROS), 2008 IEEE/RSJ International Conference on. IEEE, 2008, pp. 3384-3391.

[15] R. B. Rusu, N. Blodow, and M. Beetz, Fast point feature histograms for 3d registration, in Robotics and Automation (ICRA), 2009 IEEE International Conference on. IEEE, 2009, pp. 3212-3217.

[16] B. Drost, M. Ulrich, N. Navab, and S. Ilic, Model globally, match locally: Efficient and robust 3d object recognition, in Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on. IEEE, 2010, pp. 998-1005.

[17] D. Liu, S. Arai, Z. Feng, J. Miao, Y. Xu, J. Kinugawa, and K. Kosuge. "2D object localization based point pair feature for pose estimation," In Robotics and Biomimetics (ROBIO), 2018 IEEE International conference on. IEEE, 2018, pp. 1119-1124.

[18] M. Sharifi, M. Fathy, and M. T. Mahmoudi, A classified and comparative study of edge detection algorithms, in Information Technology: Coding and Computing, 2002 IEEE International Conference on. IEEE, 2002, pp. 117-120.

[19] J. Yang, B. Price, S. Cohen, H. Lee, and M. H. Yang, Object contour detection with a fully convolutional encoder-decoder network, in Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on. 2017, pp. 193-202.

[20] G. Bertasius, J. Shi, and L. Torresani, Deepedge: A multi-scale bifurcated deep network for top-down contour detection, in Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on. 2016, pp. 4380-4389.

[21] E. SimoSerra, S. Iizuka, K. Sasaki, and H. Ishikawa, Learning to simplify: fully convolutional networks for rough sketch cleanup, in ACM Transactions on Graphics (TOG), vol. 35, no. 4, p. 121, 2016.

[22] M. Li and K. Hashimoto, Curve set feature-based robust and fast pose estimation algorithm, Sensors, vol. 17, no. 8, p. 1782, 2017.

[23] P. J. Besl and N. D. McKay, Method for registration of 3-d shapes, in Sensor Fusion IV: Control Paradigms and Data Structures, vol. 1611. International Society for Optics and Photonics, 1992, pp. 586-607.

[24] D. Azar, "Hilditch's Algorithm for Skeletonization" Available at http://cgm.cs.mcgill.ca/ godfried/teaching/projects97/azar/skeleton.html, Jan. 2019.