Two-Stage Picking Method for Piled Shiny Objects

Yuya Sato[1], Kensuke Harada[1,2], Nobuchika Sakata[3]
, Weiwei Wan[1,2], and Ixchel G. Ramirez-Alpizar[1]

[1] Osaka University, Toyonaka 560-8531, Japan
[2] National Inst. of Advanced Industrial Science and Technology, Tsukuba 305-8651, Japan
[3] Nara Inst. of Advanced Science and Technology, Ikoma, Japan

Abstract. In this paper, we propose a novel two-steps algorithm for randomized bin-picking. Since it is difficult to detect the pose of a shiny object randomly piled in a bin, we give up picking objects one by one based on the visual information on the pile. Rather, a robot first roughly picks some of the objects from the pile without using the visual information and roughly place them onto a working table. Then, a robot picks the objects from the working table by detecting their 2D position based on the 2D RGB image. We performed experiments for multiple shiny objects with different shape and weight. Throughout the experimental study, we show that, just by adding one more step of robot motion, we can realize robust bin-picking for shiny objects.

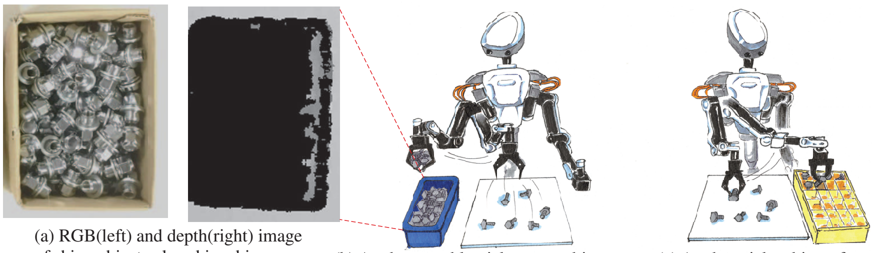
Keywords: Robotic Bin-picking, Shiny Object, Trajectory Planning

1 INTRODUCTION

Randomized bin-picking is one of the production processes where automation is most required. Once robotic randomized bin-picking is introduced to a robotic assembly process, we do not need to prepare any part-feeding machines or human workers to once arrange the parts to be picked by a robot. While a number of researches have been done on robotic bin-picking such as [1–4] and some trial has been attempted to introduce the robotic bin-picking to real production processes such as [5], robotic bin-picking is still difficult and has not been widely introduced.

One of the difficulties of robotic bin-picking lies in the limitations of vision sensor and its image processing. In typical robotic bin-picking methods [3, 6, 4, 7], a robot picks an object from the pile where its pose is estimated from image of the pile. In this method, the success rate of picking tasks depends on the accuracy of pose estimation. However, when a robot tries to pick a shiny object from the pile, it becomes difficult to obtain depth image of the pile with less chipped area by using a 3D depth sensor (Fig. 1). In this case, it becomes almost impossible to estimate the object pose.

To cope with the problem of picking shiny objects from the pile, we propose a novel approach where we do not use any visual information on the pile. The overview of our proposed method is shown in Fig. 1. Instead of picking objects one by one, our proposed method first roughly picks some of the objects without using visual information on the pile and then roughly places them onto a horizontally flat table. Once objects are placed on a table, its 2D position can be estimated just by using the 2D RGB image which



(a) RGB(left) and depth(right) image of shiny objects placed in a bin

(b) A robot roughly picks some objects and puts them onto a table (Step 1)

(c) A robot picks objects from a table

Fig. 1. Overview of the proposed method.

can be easily obtained even for shiny objects. Finally, a robot can pick the predefined number of objects from the table and put them into a returnable box. In order to pick objects from a table, we consider obtaining multiple regions on the table where the objects exist. We consider forming a graph of the end-effector path by connecting each region defined on the table. Finally, the end-effector path can be obtained by searching this graph. We performed experiments for multiple shiny objects with different shape and weight. Throughout the experimental study, we show that, just by adding one more step of robot motion, we can realize robust bin-picking for shiny objects.

The rest of the paper is organized as follows. After showing related works in Section 2, we explain our proposed method in Section 3. Finally, we show experimental results in Section 4.

2 RELATED WORKS

A number of researches have been done on randomized bin-picking. So far, the research has been mainly done on pose estimation and image segmentation of the pile [1]. Some researchers [3, 6, 7, 4] proposed bin-picking method where a robot first estimates the objects' poses by using the ICP (Iterative Closest Point) method for a given segmented depth image of the pile and then picks one of the objects from the pile by using a grasp planner. Domae et al. [2] proposed the graspability index for randomized bin-picking without estimating the objects' poses.

Recently, for the purpose of robustly performing the robotic bin-picking, learning based methods have been proposed [7–10]. However, the above methods based on a depth image of the pile cannot be applied for piled shiny objects.

So far, there have been researches of pose estimation of shiny objects such as [11]. There are some commercially available 3D vision sensors for shiny objects such as [12, 13]. However, the pose estimation of randomly stacked shiny objecta is still difficult and robust algorithm has not been investigated.

3 PROPOSED METHOD

3.1 Problem Definition

Let us consider a situation where multiple same objects are randomly stored in a bin which we call the storage box. We assume that the objects are densely stacked such that

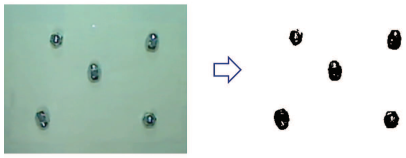


Fig. 2. RGB image (left) and its binarized image (right) of a few shiny objects placed on a table

the bottom surface of the storage box cannot be seen. This paper considers the problem where a robot picks objects from the storage box and puts the predefined number of objects into another box named the returnable box. This research does not specify the position and orientation of objects stored in the returnable box. We also assume a horizontally flat working table such that a robot can once place some of the objects onto this table. The poses of the storage box, the returnable box, and the working table are assumed to be known.

In this research, a robot does not pick objects one by one or does not simultaneously pick the predefined number of objects from the bin. Rather, a robot first roughly picks some of the objects from the bin and releases them onto the table. Then, a robot picks the predefined number of objects from the table and puts them into a returnable box. In the following, we will detail each part of the proposed algorithm.

3.2 Overview of Object Detection

This subsection first motivates our proposed two-stage method for randomized bin-picking. It is difficult to obtain a depth image of piled shiny objects with less chipped area as shown in Fig. 1. Although a 2D RGB image of such objects can be obtained, it is not easy to obtain the pose of each object from this RGB image especially when there is no texture on the object surface since we have to segment an RGB image of the pile into RGB images of each object. Fig. 1(a) shows an example of RGB and depth images of piled shiny wheel nuts. In this research, due to the difficulty of pose detection of piled shiny objects, we give up picking objects one by one. Rather, we consider a robot first roughly picking some of the objects from the pile and then releasing them above the table. In this case, we can expect to realize a configuration of objects where the objects are not crowded on a table. Fig. 2 shows an RGB image (left) and its binarized image (right) of 5 wheel nuts released above a table. We can expect from this binarized image that, as far as the color of objects can be differentiated from the color of the table, the 2D position of each object placed on a table can be estimated unless the objects are crowded on a table.

3.3 Picking Strategy from Storage Box

This subsection explains the first phase of our picking where a robot picks objects from a bin. In this phase, we do not use any visual information on the pile. On the other hand, we assume that a robot has a sensor for detecting the contact between the hand and objects. A robot first moves its hand above the bin, and then move the hand in the vertically downward direction until the contact between the hand and objects can be established. Once the contact is established, a robot tries to grasp some of the objects piled in a bin. Here, we do not impose an exact number of objects to be simultaneously picked from a bin. We only require the number of objects to be larger than the number of objects put in a returnable box. To satisfy this requirement, we can assume a variety of grippers. For example, if we use a vacuum gripper such as [14], we will adjust its suction force such that the number of picked objects becomes larger than the number of objects put in a returnable box. If we use a magnet gripper, we will adjust its magnetic force. On the other hand, if we use a multi-fingered hand, we have to design the hand itself and carefully plan its grasping strategy.

After picking objects from a bin, a robot moves the gripper above a horizontally flat table and then releases the objects. To avoid the objects densely placed on a table, the gripper has to keep a certain vertical distance from the table where the distance will be determined through experiment. Here, whether or not the objects are densely placed on a table also depends on the object's physical parameters such as shape and mass. In section 4, we will confirm whether or not the objects are densely placed on a table by using multiple objects with different physical parameters.

3.4 Picking Strategy from Table

This subsection explains the second phase of our picking strategy where a robot simultaneously picks multiple objects from a table. We set the desirable number of objects picked from a table to equal to the number of objects put in a returnable box.

Object Detection As explained in the subsection 3.2, we first capture an RGB image of objects placed on a table. We then consider subtracting the background of this image by using the image captured just before the objects are put on a table (Fig. 3-1). We furthermore obtain the region where the objects exist on a table by binarizing this image (Fig. 3-2). However, since halation often occurs in the image of shiny objects, the binarized image is often incomplete especially in the middle of each object due to the effect of the light reflection (Fig. 3-3). To cope with the problem of halation, we first diate the object region and then erose it (Fig. 3-4).

Here, each region includes a few objects while we expect that most region includes a single object if the objects are well isolated by using the method explained in the previous subsection. We estimate the number of objects included in each region by measuring its area (Fig. 3-5). The number of objects included in each region is estimated by using the following method. In advance of picking objects, we consider putting a single object on a table by our hand and measuring the object area included in its binarized image. We consider iterating this operation for several times with several different object poses. Then, we consider defining the area of a single object by averaging the area obtained through multiple measurements. When a robot releases multiple objects onto a table, the number of objects included in each region of the binarized image is estimated by dividing each area by the area of a single object. While this is an approximated method, we will check how accurate our estimation is in the following section. Also, We will take into account a situation where the number of objects is not correctly estimated. If the number of picked objects is smaller than the number of objects put in a returnable

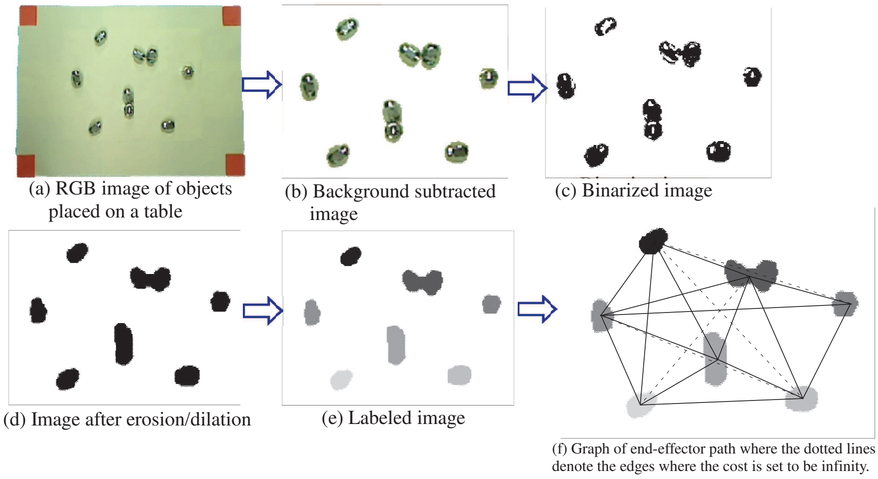


Fig. 3. Visual processing and end-effector path construction

box, we can additionally pick remaining number of objects from the table and put again in the same returnable box.

End-effector Path Then, we show a method for putting the desired number of objects into a returnable box. Our proposed method generates a graph and makes the end effector trace the shortest path of the graph. In this section, we assume the end effector of a robot to be a vacuum or a magnet gripper. Fig. 3(f) shows a graph of the end-effector path corresponding to the example of Fig. 3.

We first define nodes of the graph of the end-effector path by using the regions of objects obtained in the subsection 3.4. Then, we consider constructing edges connecting every two nodes of the graph by using a line segment where its cost is defined by the actual length of the edge. We further check the Euclid distance between each edge and each node of the graph. If an edge is close enough to a node, gripper's suction force or magnetic force will affect the objects included in the node when the end effector passes the edge. To avoid such situation, we consider setting the cost of such edges to be infinite.

Now, we explain a method for obtaining a path in this graph. We first randomly selecting a set of nodes. If the sum of the number of objects included in each node is same as the number of objects to be put in a returnable box, we consider applying this set of nodes. Then, we randomly setting an ordered list of selected nodes and calculate the total cost of the path when visiting the nodes by this order. We consider iterating this operation for a certain number of times and updating the solution path if the total cost of the path obtained through this operation is smaller than the current solution path.

4 Experiment

Fig. 4 shows the objects used in the experiment. We used steel nuts with metallic paint (Object-1 and Object-2), S-shaped steel hooks (Object-3), U-shaped black-shiny parts

Fig. 4. Objects used in experiment

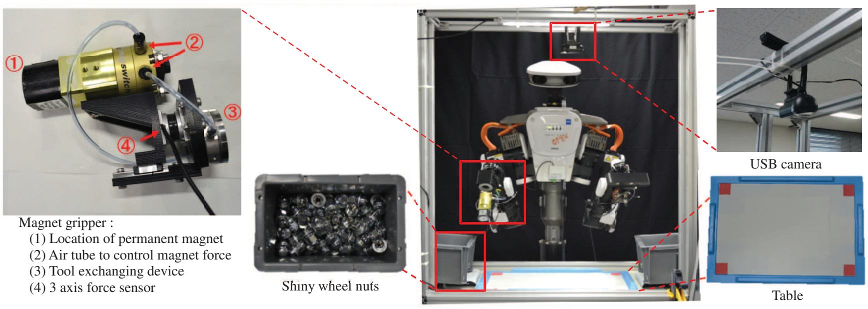


Fig. 5. Overview of experimental setup

(Object-4) and black bolts (Object-5) where the weight of each object is Object-1:40[g], Object-2:35[g], Object-3:10[g], Object-4:103[g] and Object-5:56[g]. The object-1 is expected to realize the highest success rate by using our proposed method since its shape is relatively simple. When the object-1 is released onto a table, the shape of the object projected onto a table can be uniquely determined in most of the cases. On the other hand, since there are multiple candidates for the shape of the object-2 projected onto a table, the estimation of object number is expected to be difficult. Since the object-3 and the object-4 have complex shape, the objects are expected to be densely placed on a table. The object-5 is not shiny and is used for the comparison study.

The overview of our experimental setup is shown in Fig. 5. We use a magnet gripper as an end effector. The magnetic force of this gripper is controlled by changing the position of a permanent magnet by the effect of air pressure. A 3 axis force sensor is sandwitched between the gripper and the tool exchanging device. We adjusted the magnetic force of the gripper such that the gripper can simultaneously lift up 10 wheel nuts (Object-1). We confirmed that the magnet force affects about 0.05[m] away from the tip of the end effector. We used this distance information when defining the edge weight of the graph used to search the trajectory of the end effector. We set the weight for an edge to be infinity if the distance from the edge to one of the node is less than 0.05[m]. We used a USB camera with 960×720[pixel] resolution to obtain an RGB image of objects placed on a table where the distance from the camera to the table is set as 0.8[m].

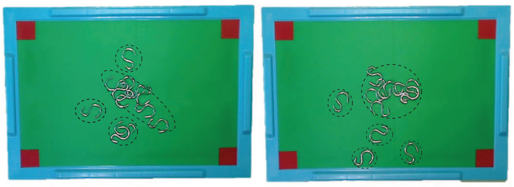


Fig. 6. Experiment on placing a set of the object-3 onto a working table

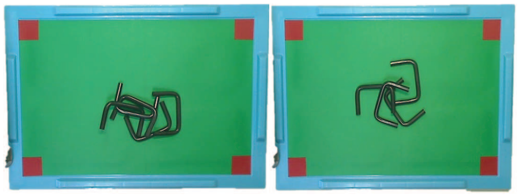
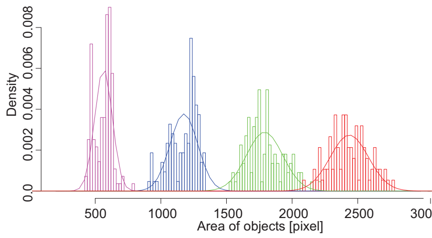


Fig. 7. Experiment on placing a set of the object-5 onto a working table

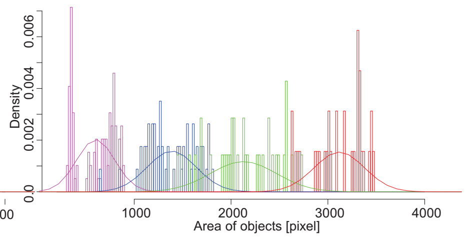
4.1 Results

Picking from Storage Box We first verified a method for picking objects from a bin and placing them on onto a table. Especially, we check if the objects are densely placed on a table depending on its shape complexity. Among objects shown in Fig. 4, we have already examined that a set of object-1 is not crowded on a table as shown in Fig. 3 and will also show that a robot can successfully perform a picking experiment in Fig. 10. In this subsection, we consider examining the object-3 and the object-4 both having complex shape whether or not a set of objects are densely placed on a table. After the gripper picks objects from a bin, the gripper releases the objects 0.1[m] above the table. After the objects are placed on a table, we captured the RGB image. We performed this experiment for 10 times for each object. Fig. 6 shows two cases of the object-3. By binarizing the RGB image, we obtained three regions including a single object and one region including two objects in the left-hand side case. We obtained four regions including a single object in the right-hand side case. On the other hand, Fig. 7 shows two cases of experiments using the object-5. By binarizing the RGB image, we could obtain just one region including all objects. Although both the object-3 and the object-5 have complex shapes, the object-5 is heavier than the object-3 and is more densely placed on a table. Hence, whether or not the objects are densely placed on a table depends on the both shape and weight of an object.

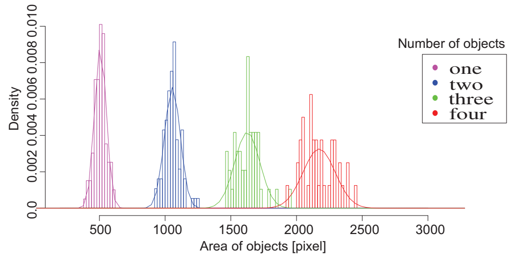
Estimation of Object Number Then, we confirm the estimation accuracy of objects' number. Fig. 8 shows histograms where the x and y axes denote the area of objects [pixel] placed on a table and its frequency, respectively. A robot picked objects from a bin and released them onto a table for 100 times. In each experiment, we obtained a binarized image of objects placed on a table and calculated each object area. We further obtained the frequency of obtaining each area. The number of objects included in each region is drawn with different colors. We can see from this figure that the number of objects can be well estimated by measuring the area for the case of the object-1 and the object-3. However, for the case of the object-2, it becomes difficult to predict the



(a) Object-1



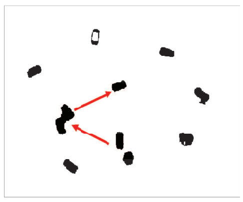
(b) Object-2



(c) Object-3

Fig. 8. Histgram of object area


(a) RGB image of objects placed on a table


(b) End-effector path when five objects are simultaneously put in a returnable box

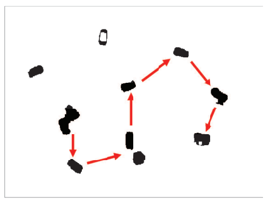

(c) End-effector path when nine objects are simultaneously put in a returnable box

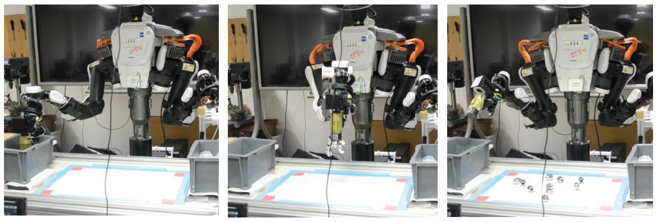
Fig. 9. End-effector path obtained from RGB image of objects placed on a table

number of objects included in each region when muliple objects are included in a single region. This is because, when the object-2 is placed onto a table, the shape of the object projected onto a table has multiple candidates.

Picking Task We performed experiments where a robot picks objects from the bin and puts the desired number of them into a returnable box. Fig. 9 shows end-effector path obtained from RGB image of objects placed on a table. This figure shows two cases where (b) a robot simultaneously picks five objects and put them into a returnable box, and (c) a robot simultaneously picks nine objects and put them into a returnable box. In both cases, the path was updated for 10 times. The calculation time of path calculation is almost 0.6[s]. The snapshot of our picking experiment by using the object-1 is shown in Figs. 10 and 11.

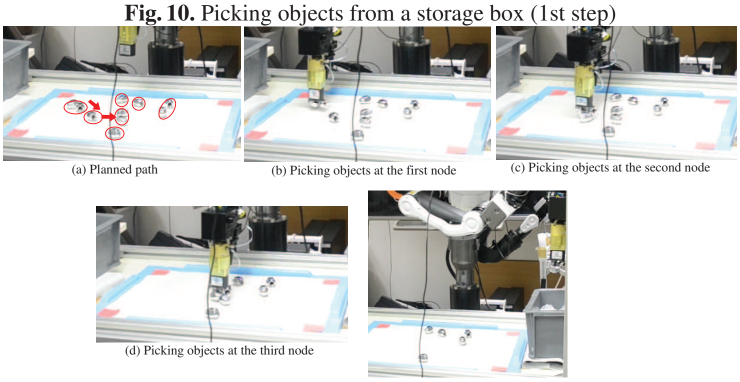
We evaluate the success rate by showing 1) success rate of number estimation of objects released on a table, and 2) success rate of picking task whether or not the desired number of objects were put in a returnable box.

By using the Object-1, Object-2 and Object-5, the success rate of number estimation is Object-1:25/25 (100%), Object-2:13/20 (65%) and Object-5:25/25 (100%). In case



(a) Picking some of the objects from randomly stacked pile (b) Releasing grasped objects on a table (c) Objects are placed on a table

Fig. 10. Picking objects from a storage box (1st step)



(a) Planned path (b) Picking objects at the first node (c) Picking objects at the second node

(d) Picking objects at the third node (e) Placing picked objects into a box

Fig. 11. Picking objects from a table (2nd step)

of the object-1, the number of objects included in each region is perfectly estimated (25/25) in spite of being shiny. On the other hand, in case of the object-2, the number of objects is correctly estimated for 13/20 times. This is because the number estimation for the region including multiple objects is not accurate. In case of the object-5, the number of objects was perfectly estimated (25/25). This is due to the result of correct binarization since the object-5 is not shiny.

The success rate of picking task is Object-1:8/10 (80%) and Object-2:6/10 (60%). In case of Object-1, a robot successfully put four objects into a returnable box for 8 out of 10 times. On the other hand, in case of Object-2, a robot successfully put four objects for 6 out of 10 times. This is due to the inaccurate number estimation of objects.

The insites obtained through our experimental study is as follows:

1) Regardless of the surface property of an object, a robot can put the predefined number of objects in a returnable box with with high success rate if the object has simple shape and if the shape projected onto a table can be uniquely determined.
2) Whether or not the released objects are crowded on a table depends on the physical property of an object such as shape, mass and friction coefficient. It the objects are often crowded on a table, we have to construct a partial graph of the end-effector path by connecting the regions on a table including a few objects.
3) If there are multiple candidates for the shape of an object projected onto a table, the estimation of object number included in each region will become inaccurate especially for a region including multiple objects. For such a case, we have to avoid the end-effector passing a region including multiple objects when solving the graph of end-effector path.

5 CONCLUSIONS

In this paper, we proposed a two-steps algorithm for randomized bin-picking. In our method, a robot first roughly picks some of the objects from the pile without using the visual information and roughly place them onto a working table. Then, a robot picks the objects from the working table by detecting their 2D position based on the 2D RGB image. We performed experiments for multiple shiny objects with different shape and weight. Throughout the experimental study, we experimented how our proposed method works by using objects with several different shapes. To improve the performance of the proposed method is considered to be our future research topic.

References

1. M. Zhou et al., "Stacked objects segmentation based on depth image", *6th Int. Conf. on Optical and Photonic Engineering*, 2018.
2. Y. Domae et al., "Fast Graspability Evaluation on Single Depth Maps for Bin Picking with General Grippers", *IEEE Int. Conf. on Robotics and Automation*, 2014.
3. D.C. Dupuis et al., "Two-Fingered Grasp Planning for Randomized Bin-Picking", *Robotics, Science and Systems 2008 Manipulation Workshop*, 2008.
4. K. Harada et al., "Project on Development of a Robot System for Random Picking–Grasp/Manipulation Planner for a Dual-arm Manipulator–", *IEEE/SICE Int. Symposium on System Integration*, 2014.
5. Fanuc Corp.,<http://www.fanuc.co.jp/ja/product/robot/baradumi.html>
6. D. Buchholz et al., "Efficient bin-picking and grasp planning based on depth data", *IEEE Int. Conf. on Robotics and Automation*, 2013.
7. K. Harada et al., "Initial Experiments on Learning-Based Randomized Bin-Picking Allowing Finger Contact with Neighboring Objects", *IEEE Int. Conf. on Automation Science and Engineering*, 2016.
8. S. Levine et al., "Learning Hand-Eye Coordination for Robotic Grasping with Deep Learning and Large-Scale Data Collection", *Int. Symposium on Experimental Robotics*, 2016.
9. K. Bousmalis et al., "Using Simulation and Domain Adaptation to Improve Efficiency of Deep Robotic Grasping", https://arxiv.org/abs/1709.07857, 2017.
10. Mahler, J. et al., "Dex-Net 2.0: Deep Learning to Plan Robust Grasps with Synthetic Point Clouds and Analytic Grasp Metrics", https://arxiv.org/abs/1703.09312, 2017.
11. Z. Wang et al., Three dimensional measurement of specular objects, Trans. of the Institute of Electronics, Information and Communication Engineers (Part D), Vol. J75-D2, No. 7, 1992 (in Japanese).
12. Canon Corp., http://cweb.canon.jp/machinevision/
13. 3D Media Company Ltd., http://www.3dmedia.co.jp/
14. T.K. Lien and P.G.G. Davis, "A novel gripper for limp materials based on lateral Coanda ejectors," CIRP Annals - Manufacturing Technology, vol. 57, 2008.