https://doi.org/10.2352/ISSN.2470-1173.2018.09.IRIACV-301

Reliable Primitive Approximation for Estimation of Robot Grasping Parameters Using 3D-DNN

Takuya TORII, Manabu HASHIMOTO; Chukyo University; Aichi, Japan

Abstract

In automatic picking by robot, it is the important to estimate the grasping parameters (grasping position, direction and angle) of the object. In this paper, we propose a method for approximating an object with primitive shape in order to estimate the grasping parameters. The basic idea of this research is to approximate the object by object primitive (hexahedron/cylinder/sphere), based on the object's surface. First, we classify the surface shape that constitutes the object using 3D-Deep Neural Network. Then, we approximate the object with object primitive using the recognition result of 3D-DNN. After that, we estimate the grasping parameters based on preset grasping rules. The success rate of approximating the object primitive with our method was 94.7%. This result is 6.7% higher than the 3D ShapeNets using 3D-DNN. Also, as an experimental result of grasping simulation using Gazebo, the success rate of grasping with our method was 85.6%.

Introduction

Recently, in international robot contest such as RoboCup [1] and World Robot Summit [2], it is being held the robotics competition assuming the home environment. In these competitions, the manipulation of everyday objects by robot is one of the tasks. In this task, it is necessary to calculate the grasping parameters with the grasping position, direction and angle of the object automatically.

As a conventional research to estimate the grasping parameters, there are methods using and not using the shape model of the object. A method that uses a shape model has the model based 3d object recognition [3, 4] used for industrial use. This method describes a feature from the shape model and the scene. And the method estimates the object's pose in the scene using these correspondence. As another approach, there is a method to calculate the parameters that the robot hand can grasp the model using a simulator such as GraspIt [5]. However, in the home environment, it is difficult to use the method that uses a shape model. On the other hand, as a method no using a shape model, there are methods to calculate the parameters by approximating the object with the primitive shape (for example, hexahedron and cylinder). Since this method can recognize the rough size and pose of the object, it has the advantage that it can be used for operation plan after grasping. As conventional method of primitive approximation, there are method that evaluating a local shape of an object [6] and evaluating a global shape of an object [7, 8, 9, 10, 11, 12]. First, the former method focuses on the difference in the surface roundness of each object primitive and approximate the object using the curvature of object. Specifically, the method approximates it using the average curvature of the object. On the other hand, the latter method approximates the object using Deep Learning [7, 8, 9, 10, 11]. This method approximate the object with primitive shape by learning the data belonging to the same primitive category. However, these methods calculate the grasping parameters using the known posture of the object. As a method of primitively approximating from the unknown posture of the object, there is a method that create a mathematical expression expressing each primitive shape and approximate the object by using matching rate between it and input data [12]. However in this method, it is difficult to apply it to point cloud data of single viewpoint because it is prerequisite to use the point cloud data of the enter perimeter.

The purpose of this research is to propose the method for approximating an object with primitive shape in order to estimate the grasping parameters without model for the object of unknown posture at a single viewpoint.

In the rest of the paper, Section 2 describe about the primitive approximation for estimating the grasping parameters of the robot. Section 3 describes the proposed method, experimental results are shown in Section 4, and we conclude the paper with a brief summary in Section 5.

Primitive Approximation for Estimating of Grasping Parameters

Type and Characteristic of Primitive Shape

In this research, we use two kinds of primitive shapes. The primitive means a simple shape that can approximate various shapes. There are Object Primitive which expresses the solid shape of an object and Surface Primitive which expresses the surface shape of an object. As the object primitive, there are hexahedron, cylinder and sphere. Nagata et al. state that many everyday objects consist of five object primitives: hexahedrons, cylinders, spheres, half semi-circular rings and half semi-rectangular rings, based on the research [13] about of how robots grasp everyday objects. From the viewpoint of versatility, we use three types of hexahedron, cylinder and sphere. In this research, we use the data viewed from a single viewpoint taken with a 3d sensor such as Kinect. Therefore, as surface primitive, there are four types of primitive surface: rectangular planes (rectangle), circular planes (disk), surfaces with linear and curvilinear components (cylindrical surface) and surfaces with curvilinear components (spherical cap).

The object primitives used in this research is different from the surface shape constituting each solid shape. For example, the hexahedron consists of a combination of rectangles, and the cylinder consists of a combination of disks and cylindrical surfaces. Figure 1 shows the relationship between the object primitive and the surface primitive. As show in Figure 1, each object primitive has the characteristics that it consists of fixed surface primitives. So, we can estimate the object primitive that apply to an arbitrary object from the type of primitive surfaces present in the object.

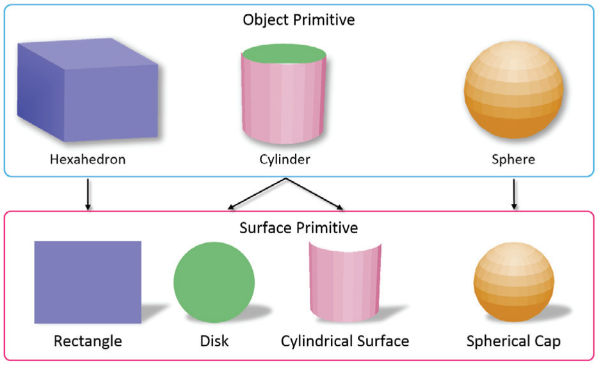

Figure 1. The relationship between object primitive and surface primitive. Object primitive has 3 types such as Hexahedron, Cylinder and Sphere. Surface primitive has 4 types such as Rectangle, Disk, Cylindrical Surface and Spherical Cap.

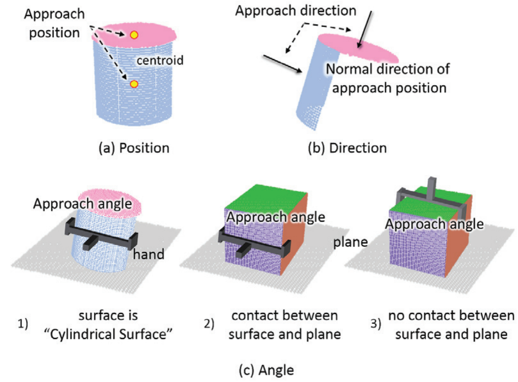

Figure 3. The diagram visualizing the estimation rule of grasping position. (a) Approach position : the centroid of a surface. (b) Approach direction : the normal direction of the approach position. (c) Roll angle : the angle perpendicular to the central axis, the angle parallel to the desk, the angle at which the width of the finger opening is the minimum.

Basic Idea

The basic idea of the proposed method is to approximate the object with object primitive based on the surface shape. First, we estimate the likelihood of each surface primitive for the surfaces of an object by using 3D-DNN. Then, we integrate each likelihood and approximate the object with object primitive using maximum likelihood of surface primitive. Figure 2 shows the diagram of basic idea. The advantage of primitive approximation using surface is the ability to estimate the object's pose from the positional relationship of surfaces and the surface shape (rectangle, disk, cylindrical surface and spherical cap).

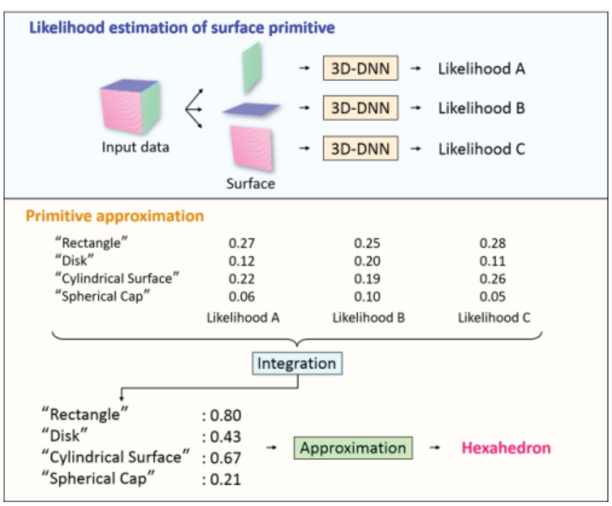

Figure 2. The diagram of the basic idea. The basic idea consists of the estimation module (top) and the approximation module (down). In the estimation module, the likelihood of surface primitives is estimated by using 3d deep learning. In the approximation module, an object is approximated with object primitive by using the result of integrating likelihoods.

Estimation Rule of Grasping Parameters

As one method of estimating the grasping parameters from an object primitive, there is a method using parameters taught to each primitive shape in advance. However, this method is not practical in the living environment such as home because the size of the object is unknown. In this research, we estimate the parameters based on the estimation rule of grasping parameters. This rule is designed to use a scene that an object is placed on a plane, approach to the surface. Figure 3 shows the diagram visualized the estimation rule of the grasping parameters. In order to pick up an object with robot hand, it is necessary to estimate the grasping parameters such as approach position, direction and angle of the hand. In this rule, the approach position is centroid point of a surface, as show in Figure 3 (a). Also, the approach direction is the normal direction of the approach position, as show in Figure 3 (b). Finally, the approach angle varies depending on the surface condition (the position, the type of surface primitive). In the case of the approach surface is Cylindrical Surface, as shown in Figure 3 (c-1), it is the angle perpendicular to the central axis. In the case of the approach surface touches the desk, as shown in Figure 3 (c-2), it is the angle parallel to the desk to avoid interference between the desk and the finger. In the case of the approach surface doesn't touch the desk, as shown in Figure 3 (c-3), it is the angle at which the width of the finger opening is minimum. The proposed method estimates the grasping parameters of the object from object primitive based on these rules.

Proposed Method
Overview

The method consists of a learning module and an approximation module, of which the algorithm flow is shown in Figure 4. In the learning module, the objects are first segmented by their surfaces using a normal distribution of point cloud data acquired by Microsoft's Kinect V2. Next, the obtained segments are learnt in the four surface primitive types using 3D-DNN. In the approximation module, first the input data is segmented by each object and the object is segmented by their surfaces. Then, the likeli-

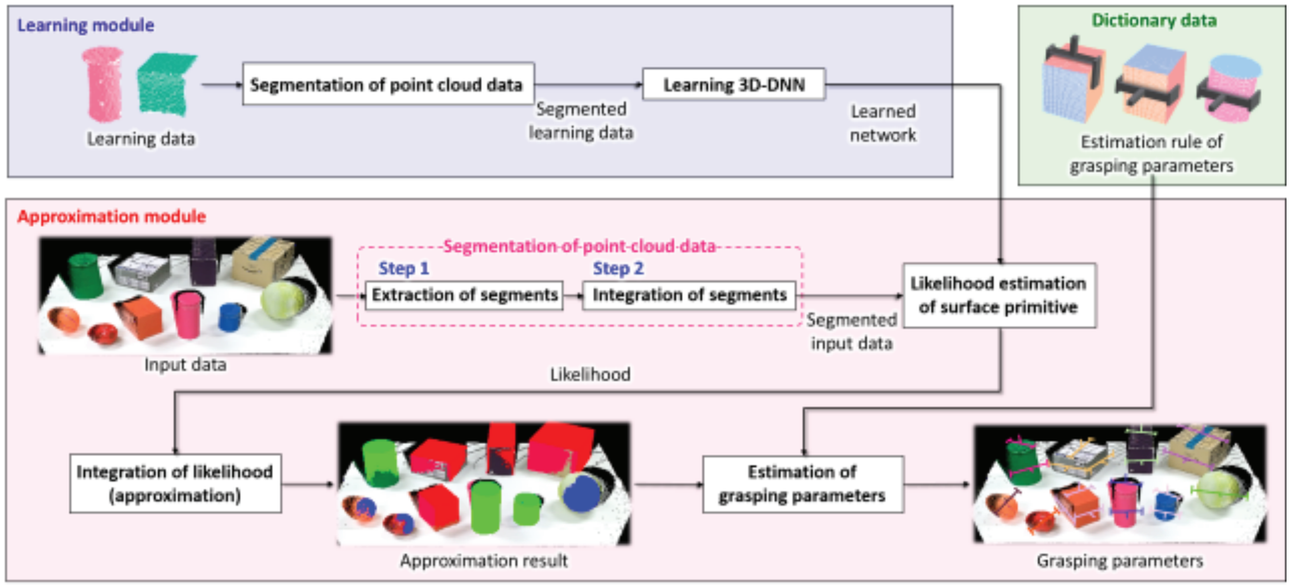

Figure 4. Flow of algorithm of the proposed method. Method consists of learning module (top) and approximation module (down).

hood of each surface primitive of the segments is estimated using the weight of network generated by learning module. After that, the likelihood is integrated for each surface primitive of the segments obtained from an object, and the object is approximated with object primitive using the maximum likelihood of the surface primitive. Finally, the grasping parameters of the object is estimated by using the estimation rule of the grasping parameters. The next section describes the segmentation of point cloud data and the 3D-DNN that was used.

Segmentation of Point Cloud Data

The proposed segmentation consists of two steps. In Step 1, segments are extracted using the method devised by Aldoma et al. [14]. A set of point cloud data assigned to the same label is called a segment. In this step, a point cloud data that exists in the vicinity and whose normal vector is the same direction are determined as the same segment. Equation 1 shows a conditional equation for determining whether two points are part of the same segment.

$$(||\mathbf{p}_h - \mathbf{p}_j|| < t_d) \wedge (\mathbf{n}_h \cdot \mathbf{n}_j < t_n) \quad (1)$$

Here, $\mathbf{p}_h$ represents the attention point in a segment, $\mathbf{p}_j$ represents the neighboring point of the attention point, $\mathbf{n}_h$ represents the average normal vector of the segment, and $\mathbf{n}_j$ represents the normal vector of the neighboring point. t_d represents the threshold for the distance from the attention point to the neighboring point, and t_n represents the threshold for the inner product of the normal vector for determining that two points are the same segment. Figure5(a) shows the segmentation result of Step 1 (the colors refer to the segment IDs). In Step 1, the average normal vector of the segment was used, so it was difficult to extract a curvilinear surface as one segment, such as a side of a cylinder.

In Step 2, the segments extracted in Step 1 were integrated. The average normal vector between the segments extracted on the side of the cylinder or sphere is different. However, since these segments are on the same surface, the normal vectors of boundaries between the segments are similar. The ratio of the number of points for the boundary between segments and the number of points with similar normal vectors was used as an index to integrate the segments. If this ratio is greater than or equal to the threshold value, the two segments are integrated. Figure 5(b) shows the segmentation result of Step 2. It was determined that the segmentation method can segment the surface of each object.

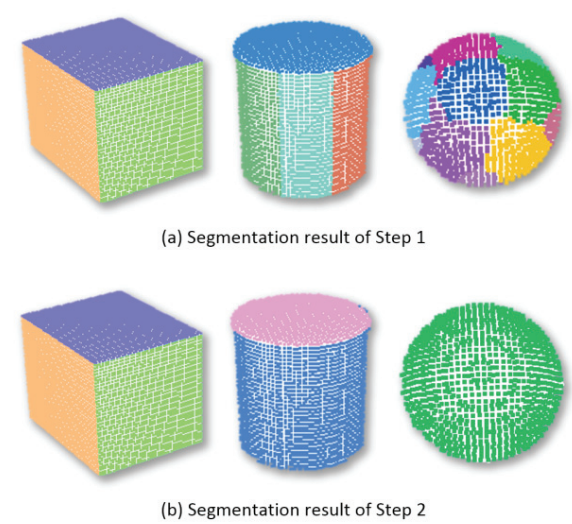

Figure 5. Segmentation result. Different colors refer to segment IDs. From left to right, objects are hexahedron, cylinder, and sphere. (a) Result of Step 1, (b) Result of Step 2.

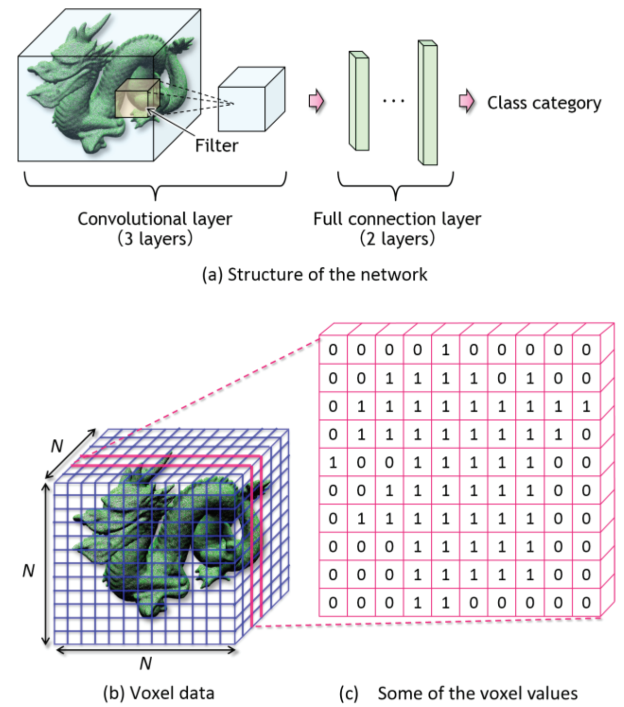

Figure 6. *Structure of 3D ShapeNets[7]. (a) The structure of the Network. (b) Transformed input data to N×N×N voxel data. (c) Voxel values of row and column of input voxel data.*

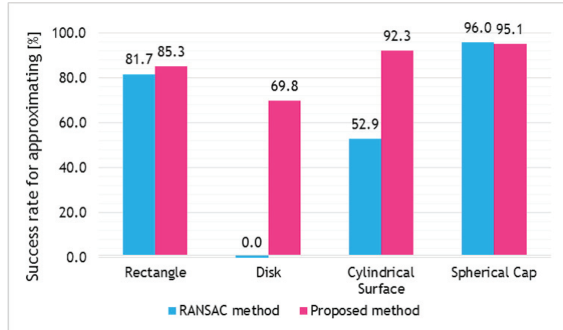

Figure 7. *Success rate for approximating with surface primitive [%]. Bule is the result of RANSAC method. Rad is the result of proposed method.*

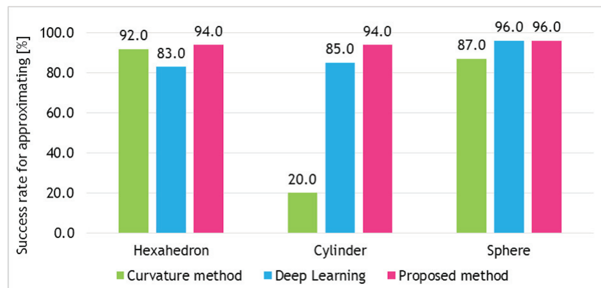

Figure 8. *Success rate for approximating with object primitive [%]. Green is the result of curvature method. Bule is the result of End-To-End Deep Learning. Rad is the result of proposed method.*

Utilized 3D DNN

3D ShapeNets, which is a 3D-deep neural network developed by Wu et al. [7] in 2015, was used in this research. Figure 6(a) shows the structure of the network of 3D ShapeNets. The network consists of three convolutional layers and two fully connection layers. Wu et al. developed the 3D convolution by transforming mesh data into voxel data. The voxel values are represented in binary. If a voxel is inside an object, it is 1. If a voxel is outside an object, it is 0. Figure 6(b)(c) shows the structure of the input data of 3D ShapeNets. The input data was transformed to N×N×N voxel data. In this research, the value of N was set to 30.

Experiments

Data Used for Approximating Experiment

To evaluate the approximate performance of the object primitive, we conducted 30 tests with hexahedron, cylinder, and sphere — shaped objects. The data is point cloud data, acquired by a Microsoft ' s Kinect V2. 10 objects per test were included. The object sizes ranged from 10×5×10∼25×17×31 (width×height×depth).

Experimental data were taken on the condition that the object is placed on a plane such as desk, and receive no occlusion or contact from sources. The input data is removed the plane which objects were placed by estimating the plane using RANSAC method as a preprocessing.

Results of Approximating Experiment

The approximate performance using both our method and comparative methods was compared on the data described in the previous section. The parameter estimation method, RANSAC, was used to approximate with surface primitive. A curvature method [6] and 3D ShapeNets [7] were used to approximate with object primitive. Figure 7 shows the success rate of approximating with the surface primitive for each method. Figure 8 shows the success rate of approximating with the object primitive for each method. As shown in Figure 7, the success rate for approximating with the surface primitive by using our method is 85.6%. This rate is 27.9% higher than with the comparative methods. As shown in Figure 8, the success rate for approximating with the object primitive by using our method is 94.7%. This rate is 6.7% higher than with the comparative methods.

The reason why our method has higher success rate for approximating than End-To-End Deep Learning is the difference in complexity of shape recognized by Deep Learning (DL). The proposed method recognizes the surface shape by DL, and the End-To-End Deep Learning recognizes the solid shape by DL. Since the surface shape is simpler in shape than the solid shape and easy to recognize, the proposed method is expected to be more accurate. Figure 9 shows the execution results of our method. Figure 9 (a) is an example of the input data. Figure 9 (b) is the result of approximating the input data with the object primitive. In this result, the point cloud data representing the object primitive is superimposed on the input data using the approximation result. The point

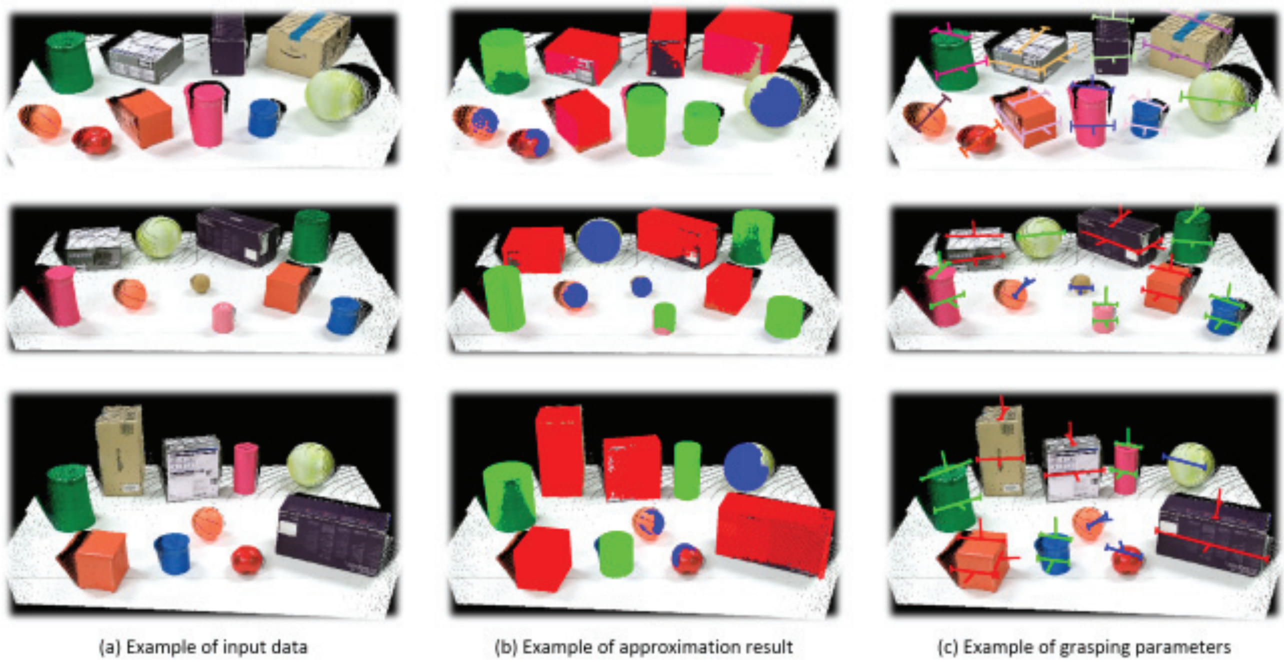

Figure 9. Execution result of method. The execution results for each input data are displayed separately for the lines. Top line shows the result of scene 1. Middle line shows the result of scene 2. Down line shows the result of scene 3. (a) Input data, (b) Approximation result of object primitive, (c) Result of estimating the grasping parameters.

cloud data of red color represents hexahedrons, the point cloud data of green color represents cylinders, and the point cloud data of blue color represents spheres. Figure 9 (c) shows the results of estimating the grasping parameters with our method. In this result, the estimated grasping parameters are superimposed on the input data. As shown in Figure 9, our method can approximate the object with object primitive and estimate the grasping parameters.

Environment Used for Grasping Experiment

For performance evaluation of grasping parameters, grasping experiment was carried out in the simulator environment using Gazebo. The robot used is the JACO2 arm of Kinova. The 3d sensor used is the SR300 of Intel. The motion path of the robot was created using the planning framework for manipulators, MoveIt! [15]. In the experiment, data on which one object was placed on the desk was used. Figure 10 shows the external view of experimental environment.

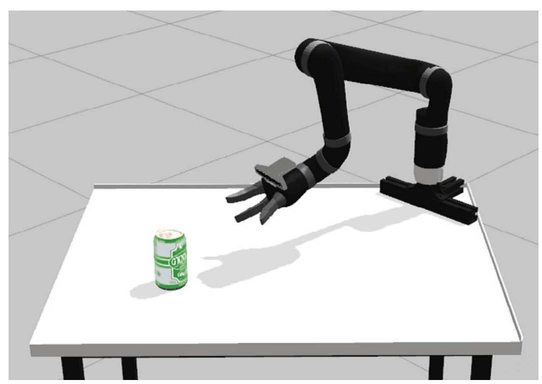

Figure 10. The external view of experimental environment. The object is placed on the desk and the arm is fixed on the desk.

Results of Grasping Experiment

The performance of the grasping parameters of the proposed method was evaluated using the environment of the previous section. The grasping success rate for each object primitive is shown in the figure 11. The success rate of grasping for hexahedron is 93.3%, the success rate of grasping for cylinder is 83.3%, the success rate of grasping for sphere is 80.0%.

A cause of failure is the approach direction is bad, primitive approximation fails. Figure 12 show the example of grasping failure. The proposed method outputs the result as shown in the fugure 12. When the height of the object is small in such a scene, the arm and the desk interfere with the grasping parameter A,B. In order to avoid interference between the arm and the desk, the grasping parameter such as C is optimum. In addition, grasping failure also occurred due to misapproximation. Because it recognized as a size larger than the actual size by misapproximation, it exceeded the maximum opening width of the arm and failed to grasp.

Conclusions

We proposed a method for approximating an object on the basis of object primitive, such as hexahedrons, cylinders, and spheres, in order to estimate the grasping parameters. The method recognizes the surface shape constituting the object using 3D-

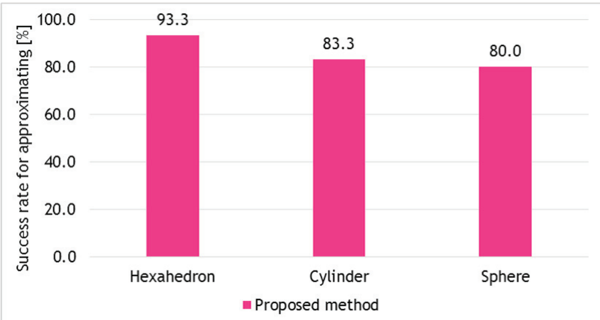

Figure 11. *The result of grasping experiment of the proposed method. Thirty scenes for each object primitive were used in this experiment. The success rate of grasping for hexahedron is 93.3%, cylinder is 83.3%, sphere is 80.0%.*

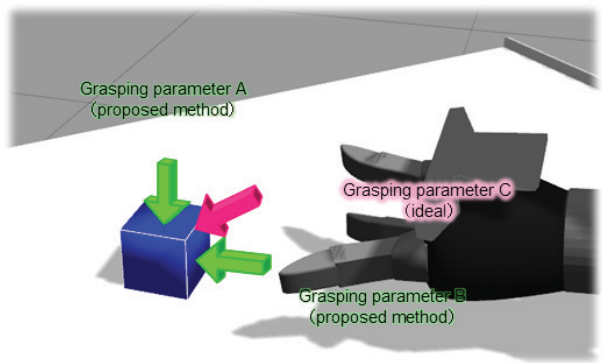

Figure 12. *The example of grasping failure.The grasping parameters A and B are the results of the proposed method. The grasping parameter C is an ideal parameter.*

DNN, and approximates with primitive shape by integrating these results. Then, the grasping parameters (the approach position, direction and angle of robot hand) of the object are estimated from the result of primitive approximation using the estimation rule of grasping parameters. The success rate for approximating with the object primitive was 94.7%, which is 6.7% higher than with comparative methods. In addition, as a result of the grasp simulation experiment using Gazebo, we confirmed that the success rate for grasping of the proposed method is 85.6%.

Acknowledgments

This work was supported by JSPS KAKENHI Grant Number JP17K06471.

References

[1] RoboCup2017, https://www.robocup2017.org/

[2] World Robot Summit, http://worldrobotsummit.org/

[3] Salti, S., Tombari, F. and Stefano, L. D.: *SHOT : Unique Signatures of Histograms for Surface and Texture Description*, Computer Vision and Image Understanding, Vol.125, pp.251-264, 2014.

[4] Drost, B., Ulrich, M., Navab, N., Ilic, S., Globally, M., Locally, M.: *Efficient and Robust 3D Object Recognition*, Proceedings of IEEE Conference on Computer Vision Pattern Recognition, pp.998-1005, 2010.

[5] Miller, A. T. and Allen, P. K.: *Graspit! A versatile simulator for robotic grasping*, IEEE Robotics & Automation Magazine, Vol.11, No.4, pp.110-112, 2004.

[6] Siddarth, J. and Argall, B.: *Grasp detection for assistive robotic manipulation*, Proceedings of IEEE International Conference on Robotics and Automation, pp.2015-2021, 2016.

[7] Wu, Z., Song, S., Khosla, A., Yu, F., Zhang, L., Tang, X. and Xiao, J.: *3D ShapeNets : A Deep Representation for Volumetric Shapes*, Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp.1912-1920, 2015.

[8] Maturana, D. and Scherer, S.: *VoxNet : A 3D Convolutional Neural Network for Real-Time Object Recognition*, Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, pp.922-928, 2015.

[9] Su, H., Maji, S., Kalogerakis, E. and Learned-Miller, E.: *Multi-view Convolutional Neural Networks for 3D Shape Recognition*, Proceedings of IEEE International Conference on Computer Vision, pp.945-953, 2015.

[10] Hegde, V. and Zadeh, R.: *FusionNet : 3D Object Classification Using Multiple Data Represen-tations*, Proceedings of IEEE International Conference on Robotics and Automation, pp.2015-2021, 2016.

[11] Riegler, G., Ulusoy, A, O., and Geiger, A.: *OctNet : Learning Deep 3D Representations at High Resolutions*, Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2017.

[12] Trung-Thien, T., Van-Toan, C. and Denis, L.: *Extraction of reliable primitives from unorganized point clouds*, 3D Research, 2015.

[13] Markus, P., Tamim, A. and Rudiger, D.: *Unions of Balls for Shape Approximation in Robot Grasping*, IEEE/RSJ International Conference on Inteligent Robots and Systems, pp.1592-1599, 2010.

[14] Aldoma, A., Tombari, F., Rusu, R. B. and Vincze, M.: *OUR-CVFH - Oriented, Unique and Repeatable Clustered Viewpoint Feature Histogram for Object Recognition and 6DOF Pose Estimation*, Proceedings of Joint 34th DAGM and 36th OAGM, pp.113-122, 2012.

[15] Chitta, S., Sucan, I. and Cousins, S.: *Moveit! [ROS topics]*, IEEE Robotics & Automation Magazine, Vol.19, No.1, pp.18-19, 2012.

Author Biography

Mr. Takuya Torii received the B.E. degrees from Chukyo University, Aichi, Japan, in 2017. Now he is a graduate student of the university, and is very interested in technologies about 3-D object recognition and its application to robot manipulation. He is a member of Japanese societies of IPSJ and JSPE.

Manabu Hashimoto received the B.E. and M.E. degrees from Osaka University, Osaka, Japan, in 1985 and 1987 respectively. He joined Mitsubishi Electric Corporation in 1987, and he has been engaged in research on robot vision, image recognition, pattern recognition, and human sensing in Manufacturing Development Laboratory, Industrial Electronics and Systems Development Laboratory, and Advanced Technology R&D Center of the company. He received Ph.D. degree from Osaka University in research on 3-D object recognition in 2000. Since 2008, he has been a professor of Chukyo University, and he is the Dean of School of Engineering. He received the Technical Innovation Award in 1998 from the Robotics Society of Japan, Excellent Academic Award in 2012 from Symposium on Sensing via Image Information, and the Best Paper Award of IWAIT2017 and IWAIT2018. He is a member of IEEE and Japanese societies of IEICE, IPSJ, IEEJ, RSJ, and others.