Detecting layered structures of partially occluded objects for bin picking

Yusuke Inagaki[1], Ryosuke Araki[2], Takayoshi Yamashita[3], and Hironobu Fujiyoshi[4]

***Abstract*— When robots engage in bin picking of multiple objects, a failure in grasping partially occluded objects may occur because other objects may overlap the desired ones. Therefore, the layered structure of objects needs to be detected, and the picking order needs to be established. In this paper, we propose a new dataset that evaluates not only the area of objects but also the layered structures of objects. In this dataset, three tasks are targeted: object detection, semantic segmentation, and segmentation of occluded areas for bin picking of multiple objects. The dataset, called "the Amazon Robotics Challenge (ARC) Multi-task Dataset" contains 1,500 RGB images and depth images, including all scenes containing bounding box labels, semantic segmentation labels, and occluded area labels. This enables representing the layered structure of overlapped objects with a tree structure. A benchmark of the ARC multi-task dataset demonstrated that occluded areas could be segmented using a Mask regional convolutional neural network (R-CNN) and that layered structures of objects could be predicted. Our dataset is available at the following URL:`http://mprg.jp/research/arc_dataset_2017_e`.**

I. INTRODUCTION

Because of continuous developments in robot technology, logistics warehouses are being automated in electronic commerce. For example, a robot can carry a shelf containing goods to the picking operator when an order of goods is inputted by the customer into the company database [1]. However, because products are currently being carried manually, automation of conveyance using picking robots is desired. A logistics warehouse has rows upon rows of multiple objects in bins, and many object are partially occluded by other products. When a robot operator conducts bin picking under such an environment, a failure may occur in attempts to grasp partially occluded objects because of overlapping of objects. The conventional picking method does not take occlusion into consideration, leading to cases of grasping a partially occluded object [12]. Bin picking of multiple objects requires detecting layered object structures and determining the picking order. Thus, occluded areas of objects and occluded objects need to be detected.

Some public datasets on bin picking by robots such as the Massachusetts Institute of Technology (MIT)'s Grasping Dataset [7] and UC Berkeley's Amazon Picking Challenge Object Scans [6] are available. MIT's dataset has segmentation labels for areas where objects can be grasped. UC Berkeley's Amazon Picking Challenge Object Scans have segmentation labels in point clouds. However, these datasets cannot evaluate occluded areas in objects and cannot be used to detect layered structures. In this paper, we propose a new data set, *the Amazon Robotics Challenge (ARC) Multi-task Dataset*, that evaluates the layered structure of objects. This data set includes the following three tasks: object detection, semantic segmentation, and segmentation of occluded areas for bin picking of multiple objects. The dataset contains 1,500 RGB and depth images and includes all scenes containing bounding box labels, semantic segmentation labels, and occluded area labels. This enables representing layered structures of overlapped objects with a tree structure. We evaluated a benchmark of *ARC Multi-task Dataset* for segmenting occluded areas and for predicting layered structures of objects.

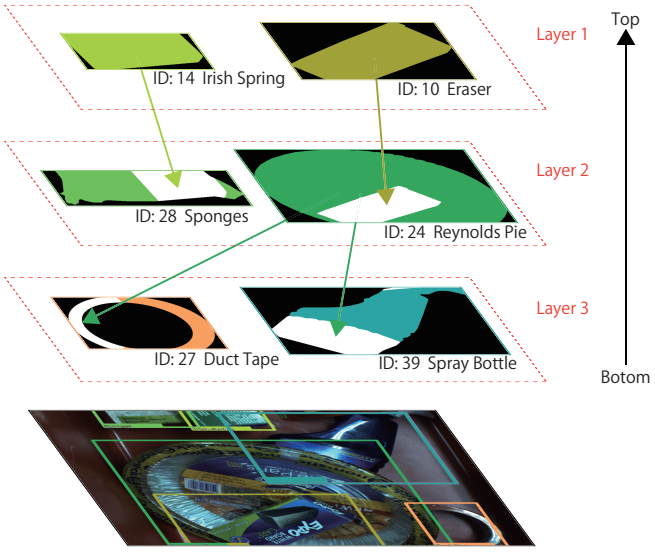

Fig. 1. Layered structure tree. The object region and the occluded area were used to determine the relationship.

Our main contributions are as follows:

1) A new dataset, *ARC Multi-task Dataset*, to evaluate the layered structures is presented.

2) We propose a novel task, occluded area segmentation using the dataset. This enables detecting layered structures of objects using a tree structure.

[1]Yusuke Inagaki is with the Dept. of Robotics Science and Technology, Grad. School of Engineering, Chubu Univ., Aichi, JP`yusuke@mprg.cs.chubu.ac.jp`

[2]Ryosuke Araki is with the Dept. of Computer Science, Grad. School of Engineering, Chubu Univ., Aichi, JP`ryorsk@mprg.cs.chubu.ac.jp`

[3]Takayoshi Yamashita is with the Faculty of Computer Science, College of Engineering, Chubu Univ., Aichi, JP`takayoshi@isc.chubu.ac.jp`

[4]Hironobu Fujiyoshi is with the Faculty of Robotics Science and Technology, College of Engineering, Chubu Univ., Aichi, JP`fujiyoshi@isc.chubu.ac.jp`

TABLE I
DATASETS COMPARISON

Dataset	#Sample	#Depth Image	#Class	#Bbox	#Semantic Segmentation Label	#Occluded Area Segmentation Label	#Grasping Label	#Robotic Grasping Dataset	#bin picking
Pascal VOC 2012 [2]	17k		20	27k	2913	-	-		
COCO [3]	330k		80	2500k	200k	-	-		
CMU Kitchen Occlusion Dataset [4]	1600		8	1600	-	-	-		
Cornell University The Grasping Rectangle Dataset [5]	1035	✓	280	-	-	-	1035	✓	
UC Berkeley APCOS [6]	600	✓	27	-	600	-	-	✓	✓
MIT Grasping Dataset(picking hand) [7]	778	✓	-	-	-	-	778	✓	✓
MIT Grasping Dataset(suction hand) [7]	1837	✓	-	-	-	-	1837	✓	✓
ARC Multi-task Dataset	1500	✓	40	14,824	1500	14,824	-	✓	✓

II. RELATED WORK

A. Dataset of object detection and segmentation

Object detection is a method for localizing an object in an image [9] [10] [11]. The location of an object is typically represented by a bounding box. A class label corresponds to the bounding box. For general objects, a public dataset called PASCAL VOC Dataset [2] has been proposed. This dataset has 20 classes, 17,000 RGB images, and more than 27,000 objects with bounding box labels.

A public dataset called the CMU Kitchen Occlusion Dataset [4] can be used for object detection targeting partially occluded objects. This dataset has eight kinds of items without texture, 1,600 RGB images, bounding box labels only for target items in all scenes, and occlusion models of arbitrary viewpoints. It includes about four to eight objects per scene for partially occluded objects and other objects. The dataset detects a partially occluded object, but it cannot be used to detect layered structures.

Semantic segmentation is a method of classifying the pixel level [13][14][15]. Object classes are labelled on pixels for labels of semantic segmentation. In contrast to a bounding box label, individual instances of objects do not need to be segmented. This enables the labeling of objects for which individual instances are hard to define, such as the sky, ground, or walls. Semantic segmentation has a publicly available large-scale dataset called the COCO Dataset [3]. It has 80 classes, 330,000 RGB images, and 200,000 semantic segmentation labels.

B. Dataset of robotic grasping

Robots need to be able to detect objects and the gripping position required, to segment the object regions, and to set a gripping order. Cornell University provides what it calls The Grasping Rectangle Dataset [5] for picking up multiple objects. It can be used by robots in detecting the grasping position when they pick up multiple objects. The dataset has 280 kinds of items, 1,035 RGB images, point group data for each image, and grasping position data labels. Each item is photographed in a plurality of directions and poses.

Public datasets for bin picking of multiple objects by robots include MIT's Grasping Dataset [7] and UC Berkeley's Amazon Picking Challenge Object Scans [6]. MIT's Grasping Dataset is used in detecting the grasping position. Grasping Dataset has two types of picking methods, one with a picking hand and one with a suction hand. It includes 778 RGB images, plus corresponding depth images and segmentation data for robot hands and 1,837 RGB images, plus corresponding depth images and segmentation data for suction hands. UC Berkeley's Amazon Picking Challenge dataset has 600 RGB images, plus corresponding depth images, point clouds, and segmentation labels for the point clouds.

A list of the data sets described so far are shown in Table I. In table I, the datasets include targets for detecting picking positions and targets for detecting object areas. Although these data sets can enable an object to be grasped, only a method using depth images can establish their grasp order, and the current ones do not consider occlusion. Therefore, such a method is necessary to predict the layered structure of an object using the object area and the occluded area.

III. ARC MULTI-TASK DATASET

A. Overview

The ARC Multi-task Dataset provides a dataset for robots to pick up items in the Amazon Robotics Challenge(ARC). The dataset contains 40 kinds of items used in the competition. Each item has various shapes and attributes such as box-shaped items, items packed in vinyl, and non-rigid items. Image samples are shown in Fig 2. The ARC Multi-task Dataset contains1280×960 pixel RGB images, depth images and 3D models for all items. Each scene has one to eight items, and the scenes are randomly generated with partial occlusion of various sizes. There are 1,100 scenes shot with a plurality of items and 10 scenes shot with only one item. The following three labels are annotated for all scenes.

1) ***Bounding box label****:* A bounding box label is used for learning and evaluating object recognition. It has text data with the coordinates for all the scenes and the ID of the item. Annotation is attached only to a range that can be confirmed by a person, and annotation is attached only

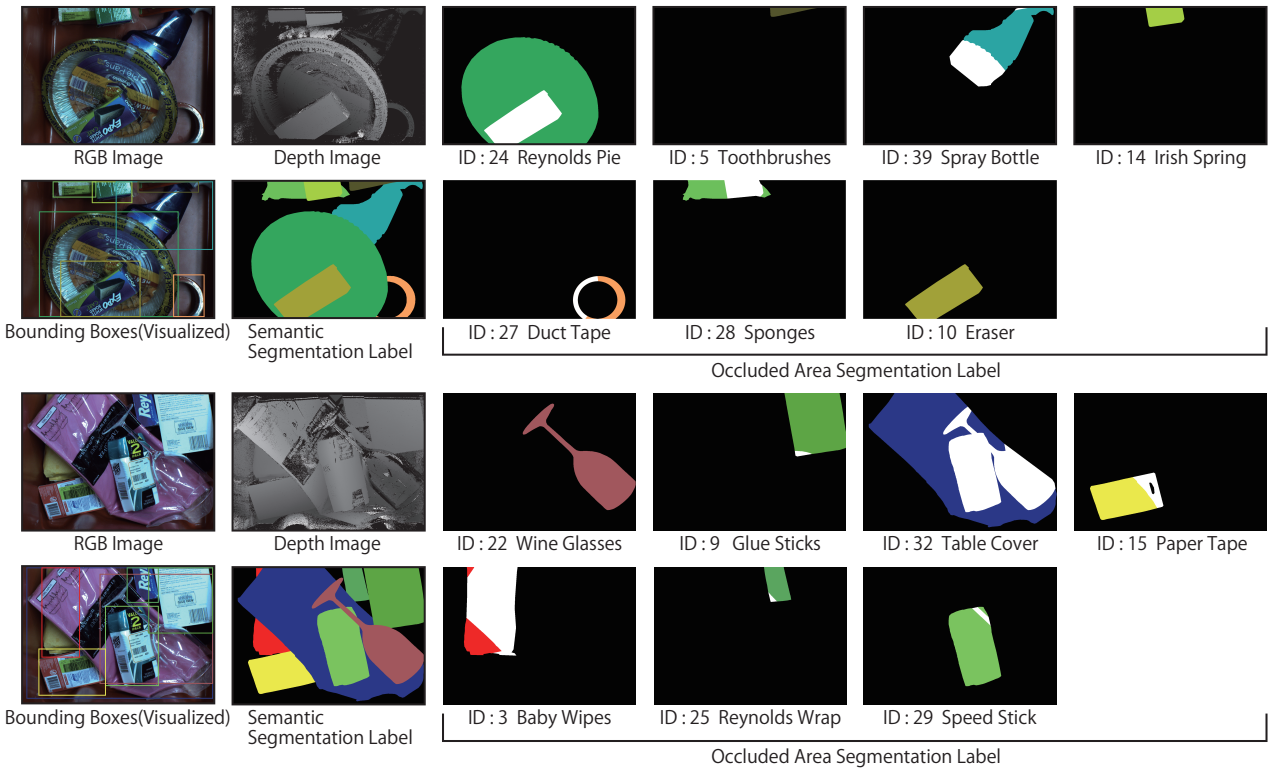

Fig. 2. Examples of ARC Multi-task Dataset

to the visible range when occlusion occurs. Bounding box labels are needed to acquire the detection instance for the robots.

*2) **Object Segmentation Label**:* An object segmentation label is used for learning and evaluating semantic segmentation of the object region. Semantic segmentation images color-coded on a pixel-by-item basis for every item region are included for all scenes. Object segmentation labels are needed to acquire the object area of the detected object.

*3) **Occluded Area Segmentation Label**:* An occluded area segmentation label is used for learning and evaluating segmentation with occluded areas and for learning and evaluating segmentation with occlusion region interpolation. In these labels, one image is held for all items in the scene, and it contains object area labels and occluded area labels. Each colored region indicates object segmentation, and each white region indicates the occluded area. The picking order needs to be acquired by predicting the layered structure of the objects.

B. Configuration task

We propose new tasks using the ARC Multi-task Dataset, occluded area segmentation, interpolation segmentation, and detecting the layered structure of objects.

*1) **Occluded area segmentation**:* Occluded area segmentation is a task of detecting the occluded area at the pixel level for each object. Occluded area segmentation acquires the occluded object and occluded area for each object. A cutout image of the detected object area is used, or the results of semantic segmentation and results of the interpolation segmentation are used as input for occluded area segmentation. In learning this type of segmentation, labels of the occluded area in Fig. 2 are used. Thus, the occluded area for each object can be learned.

In occluded area segmentation, the layered structure can be detected if each object's occluded area can be detected correctly even in a small area. For this reason, we use the occlusion undetected rate and the false positive rate for each object in evaluating the occluded area segmentation. Also, class accuracy and Mean IoU are used to evaluate occluded area segmentation.

The undetected rate of occlusion is shown in Equation 1, and the false positive rate of occlusion is shown in Equation 2.

$$Undetected\ rate = \frac{O_u}{o} \tag{1}$$

$$False\ positive\ rate = \frac{O_f}{n} \tag{2}$$

The undetected rate is the rate at which the occlusion occurred area cannot be predicted. Therefore, let O_u be the number of undetected objects in Equation 1 and o be the total number of occlusion occurring objects. The false positive rate is the rate of cases in which the occluded area is predicted

Algorithm 1 Creating a vertical relationship layer

Input: list of occlusion segmentations of each object: $X = \{x_1, \ldots, x_n\}$
Output: list of tree layers: L
Initialize result layers: $L \leftarrow \{\}$
Initialize a list for depth 1: $l_1 \leftarrow \{\}$
for $x_i \in X$ **do**
    if x_i does not contain occluded areas **then**
        Add x_i into l_1: $l_1 \leftarrow l_1 \cup \{(x_i, \text{None})\}$
        Remove x_i from X: $X \leftarrow X \backslash \{x_i\}$
Add l_1 into L: $L \leftarrow l_1$
Set depth d as 2: $d \leftarrow 2$
while $X \neq \emptyset$ **do**
    Initialize a list for depth d: $l_d \leftarrow \{\}$
    for $x_i \in X$ **do**
        if The occluded area is covered by objects in L **then**
            Find upper object x_j on x_i
            Add x_i into l_d: $l_d \leftarrow l_d \cup \{(x_i, x_j)\}$
            Remove x_i from X: $X \leftarrow X \backslash \{x_i\}$
    Add l_d into L: $L \leftarrow l_d$
    $d \leftarrow d + 1$

as an incorrect area for an occluded object or in which the occluded area is predicted in a non-occurring object. Therefore, let O_f be the number of erroneously detected objects in Equation 2 and n be the total number of objects.

*2) **Interpolation segmentation**:* Interpolation segmentation is a task of detecting the object area in the occluded area at the pixel level for each object. This method acquires the object area interpolation in the occluded area for each object. In learning this type of segmentation, labels for the object area and the occluded area in Fig. 2 are used. Global accuracy, class accuracy, and mean IoU are used to evaluate the interpolation segmentation.

*3) **Detecting layered structures**:* Detecting layered structures is a task of creating a layered structure tree used for all objects in an image. Doing this requires detecting objects, occluded objects, and occluded areas. Therefore, occluded area segmentation is used for each object to acquire the occluded area. Occluded objects, with overlapping in the occluded area are necessary for predicting layered structures. The creation algorithm of the layered structure tree is shown in algorithm1. x_i is the occluded area segmentation and the result for each object, $X = \{x_1, \ldots, x_n\}$ is the occluded area segmentation and the result from the image, and L is the layered structure tree using the algorithm. The uppermost object in the layered structure tree is considered to be layer 1, and the underlying object is considered to be layer 2. The layered structure tree can grasp the layered structure and acquire the gripping order. An example of creating a layered structure tree of objects is shown in Fig1. The evaluation used the correct answer rate. This rate is the proportion of objects that can predict the layered structure correctly in a layered structure tree. Let T_t be the number of objects in which the layered structure of expression 3 is correct.

$$Tree\ correct\ answer\ rate = \frac{T_t}{n} \tag{3}$$

IV. Estimate the layered structure tree

This section describes detecting the layered structure tree using the ARC multi-task dataset. The network configuration is shown in Fig. 3.

A. Occluded Area Mask R-CNN

Creating the layered structure tree requires acquiring the position of the object, the object area, and the occluded area. Therefore, we used a network with an occluded area segmentation task added to a Mask R-CNN [8]. The network (A) in Fig. 3 obtains segmentation results for the occluded area by adding a deconvolution network that performs occluded area segmentation based on the Mask R-CNN.

B. Interpolation Mask R-CNN

Our method of acquiring the occluded area uses that area with a visible area of the object and the object area interpolation of the occluded area. We use two networks, a network in which an interpolation segmentation task is added to the Mask R-CNN and a network for performing occluded area segmentation using the two results. The network (B) in Fig. 3 obtains segmentation results for the interpolation by adding a deconvolution network that performs occluded area segmentation to interpolate the segmentation area based on the Mask R-CNN. The occluded area can be obtained by subtracting the object area from the results of interpolation segmentation.

V. Experiment

We conducted an experiment to evaluate the effectiveness of the occluded area Mask R-CNN and the Interpolation Mask R-CNN learned using the ARC multi-task dataset. In the experiments, we evaluated the segmentation object areas, interpolation segmentations, occluded area segmentations, and layered structure trees that could be acquired by each method.

A. Experiments Evaluating the masks, interpolation masks, and occluded area masks

These experiments were done to evaluate the object area, interpolation segmentation, and occluded area segmentation. From Table II, the occluded area Mask R-CNN and interpolation Mask R-CNN could segment the object area with the same precision as that of the Mask R-CNN.

Figure 4 shows an example of the output results of the interpolation segmentation. Interpolation segmentation could handle the interpolate occluded area in the crayons and the laugh out loud jokes in Fig. 4. Also, cases of large occlusion occurring at the end of objects as shown in the green book of Fig. 4 were difficult to interpolate, and accuracy was reduced. However, interpolation was possible.

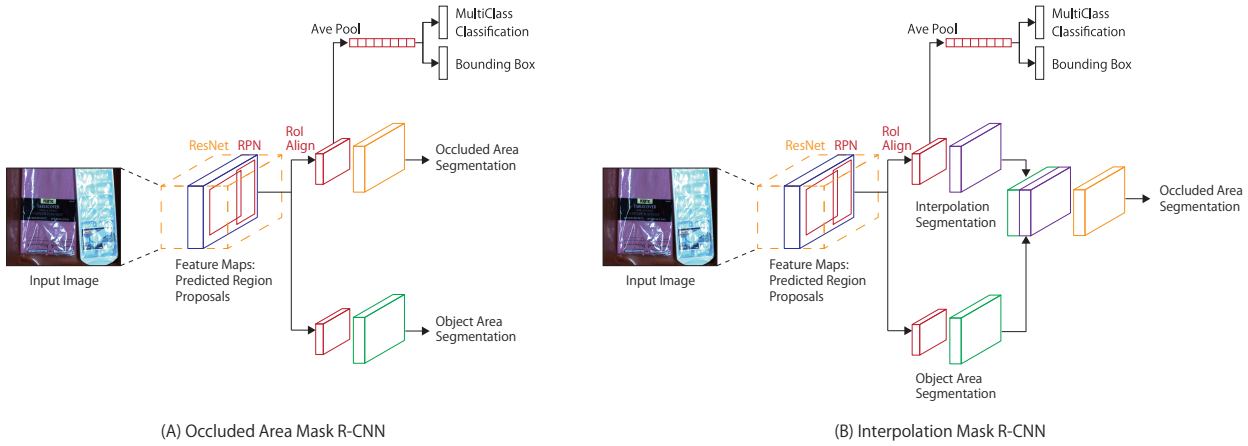

Fig. 3. Network configuration. Obtaining an object category, object position, object area, and occluded area, which are the results of occluded area Mask R-CNN with respect to the detected object.

TABLE II
EVALUATION LIST

	Object Area			Interpolation			Occluded Area				Layered Structure Tree
	Global Accuracy	Class Accuracy	Mean IoU	Global Accuracy	Class Accuracy	Mean IoU	Undetected rate	False positive rate	Class Accuracy	Mean IoU	Tree Correct answer rate
Mask R-CNN	76.7	61.8	54.0	-	-	-	-	-	-	-	-
Occluded Area Mask R-CNN	**80.2**	65.3	**56.9**	-	-	-	**16.4**	**16.1**	**45.9**	**36.6**	**68.8**
Interpolation Mask R-CNN	79.4	**68.1**	56.1	**80.6**	**63.7**	**51.8**	27.5	25.7	28.0	17.9	41.9

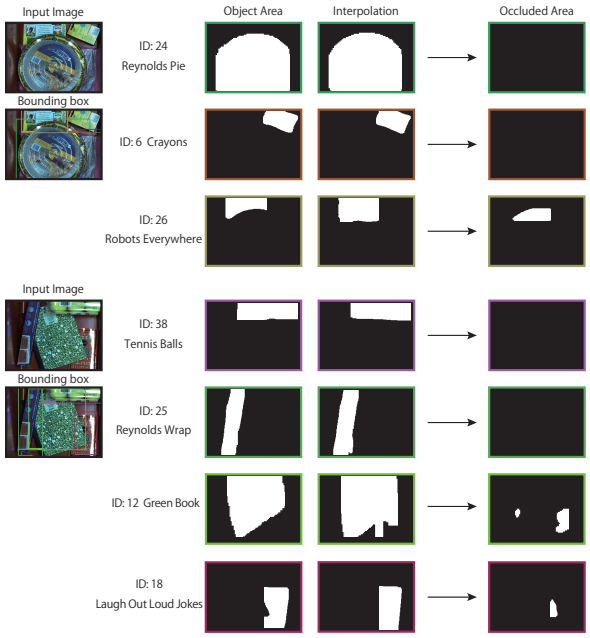

Fig. 4. Example of Interpolation Mask R-CNN

Figure 5 shows an example of the output results of the occluded area segmentation. Table II shows the results for occluded areas by Mask R-CNN, occluded area Mask R-CNN, and interpolation Mask R-CNN. The occluded area precision of the interpolation Mask R-CNN was lower than that of the occluded area Mask R-CNN in all cases. The accuracy of occluded area segmentation in the interpolation Mask R-CNN is considered to be low because it depends on the accuracies of the object area segmentation and the interpolation segmentation. However, interpolation Mask R-CNN should be able to detect the occluded area with high precision when higher precision is achieved with object area segmentation.

B. Detection of layered structure tree

The layered structure tree of overlapping objects could be obtained using the results of object area segmentation and the results of occluded area segmentation. Fig. 5 shows a layered structure tree of objects created using the occluded area Mask R-CNN. Table II reveals that the occluded area Mask R-CNN could create a layered structure with the highest accuracy. It could create a layered structure correctly, enabling gripping to be performed for any object. However, many cases erroneously predicted a fine occluded area in an incorrect area. For this reason, many cases with no object in Layer 1 occurred, and a layered structure tree could not be created. Our approach can create a layered structure tree for an uncomplicated scene where large occlusion occurs in each object.

VI. CONCLUSION

In this paper, we proposed the ARC Multi-task Dataset, which enables predicting the occluded areas of objects. The dataset had an undetected rate of only 16.4% and a false positive rate of 16.1% when detecting occlusion occurrence regions. A layered structure tree could be created with an

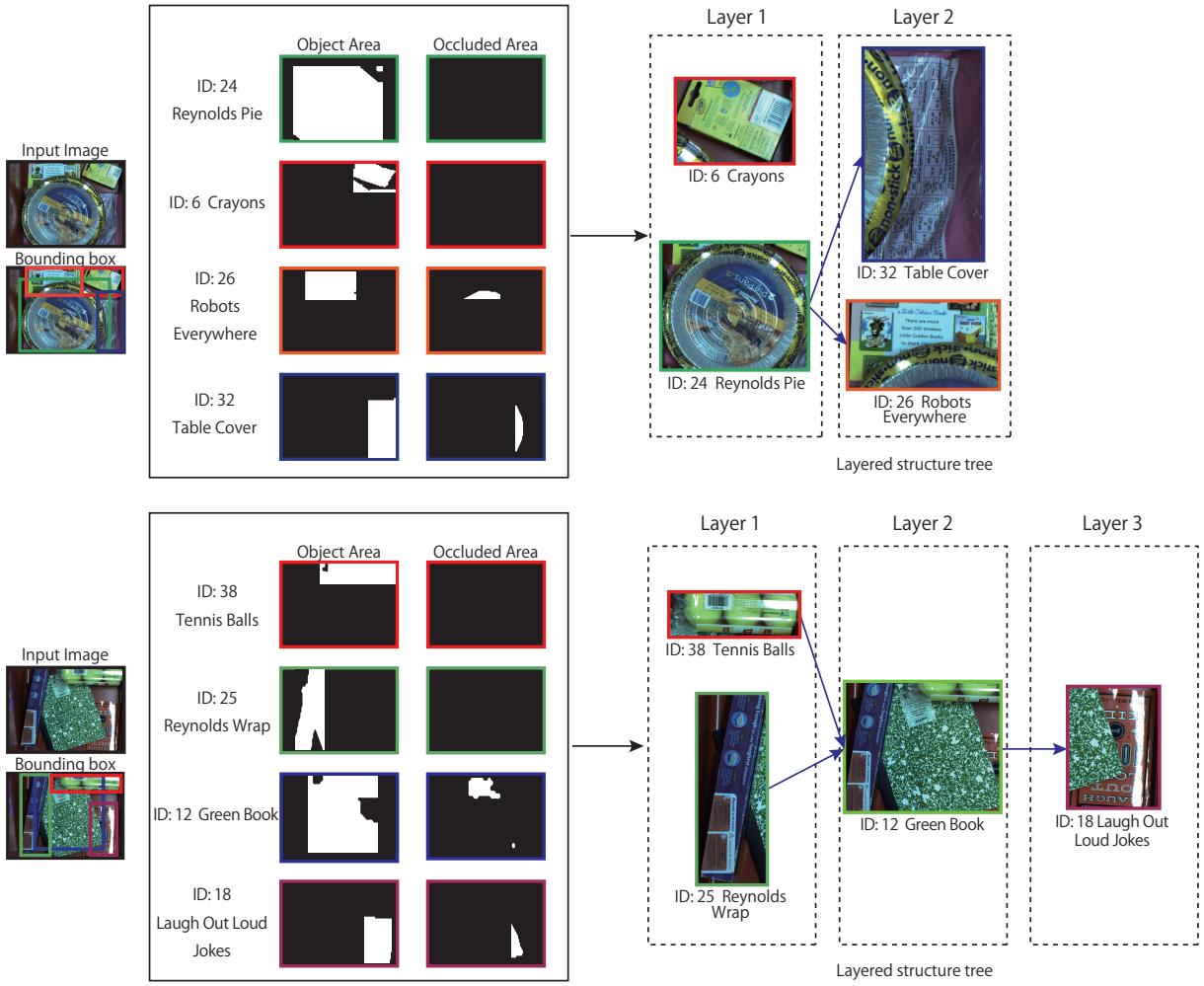

Fig. 5. Examples of Occluded Area Mask R-CNN and layered structure tree

accuracy of 68.8%. Future work includes acquiring a more accurate layered structure tree and experiments using actual machines.

Acknowledgement. This work was carried out thanks to the budget provided for the NEDO "next generation artificial intelligence & robot core technology development."

REFERENCES

[1] Amazon Robotics, "Vision", [Online]https://www.amazonrobotics.com/\#/vision
[2] M. Everingham, S. M. A. Eslami, L. Van Gool, , C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal Visual Object Classes Challenge: A Retrospective," IJCV, 2015.
[3] Tsung-Y. Lin, M. Maire, S. J. Belongie, L. D. Bourdev, R. B. Girshick, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common Objects in Context," ECCV, 2014.
[4] E. Hsiao, and M. Hebert, "Occlusion reasoning for object detection under arbitrary viewpoint," CVPR, 2012.
[5] I. Lenz, H. Lee, and A. Saxena, "Deep Learning for Detecting Robotic Grasps," RSS, 2013.
[6] UC Berkeley, "Amazon Picking Challenge Object Scans,"[Online] http://rll.berkeley.edu/amazon_picking_challenge/
[7] A. Zeng, S. Song, K. T. Yu, E. Donlon, F. R. Hogan, M. Bauza, D. Ma, O. Taylor, M. Liu, E. Romo, N. Fazeli, F. Alet, N. C. Dafle, R. Holladay, I. Morona, P. Q. Nair, D. Green, I. Taylor, W. Liu, T. A. Funkhouser, and A. Rodriguez, "Robotic Pick-and-Place of Novel Objects in Clutter with Multi-Affordance Grasping and Cross-Domain Image Matching," ICRA, 2018.
[8] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," ICCV, 2017.
[9] S. Ren, K. He, R. B. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," CVPR, 2015
[10] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. E. Reed, C. Y. Fu, and A. C. Berg, "SSD: Single Shot MultiBox Detector," ECCV, 2016.
[11] J. Redmon, S. K. Divvala, R. B. Girshick, and A. Farhadi, "You Only Look Once: Unfied, Real-Time Object Detection," CVPR, 2015.
[12] Y. Domae, H. Okuda, Y. Taguchi, K. Sumi, and T. Hirai, " Fast graspability evaluation on single depth maps for bin picking with general grippers," ICRA, 2014.
[13] V. Badrinarayanan, A. Kendall, R. Cipolla, and S. Member, "SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation," CVPR, 2015.
[14] J. Long, E. Shelhamer, and Trevor Darrell, "Fully Convolutional Networks for Semantic Segmentation," CVPR, 2014.
[15] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," CVPR, 2015.