

# Instance Segmentation of Visible and Occluded Regions for Finding and Picking Target from a Pile of Objects

Kentaro Wada, Shingo Kitagawa, Kei Okada and Masayuki Inaba

University of Tokyo, JSK Laboratory

{wada, s-kitagawa, okada, inaba}@jsk.imi.u-tokyo.ac.jp

**Abstract**—We present a robotic system for picking a target from a pile of objects that is capable of finding and grasping the target object by removing obstacles in the appropriate order. The fundamental idea is to segment instances with both visible and occluded masks, which we call ‘instance occlusion segmentation’. To achieve this, we extend an existing instance segmentation model with a novel ‘relook’ architecture, in which the model explicitly learns the inter-instance relationship. Also, by using image synthesis, we make the system capable of handling new objects without human annotations. The experimental results show the effectiveness of the relook architecture when compared with a conventional model and of the image synthesis when compared to a human-annotated dataset. We also demonstrate the capability of our system to achieve picking a target in a cluttered environment with a real robot.

## I. INTRODUCTION

With recent progress in deep learning, especially convolutional neural networks, the robotics community has improved the ability of the robot to find and pick various target objects in clutter [1]–[5], even when including novel objects [6]. However, these works restrict the environment to have few occluded target objects. Our goal in this work is to develop a framework that enables a robot to pick various target objects in the appropriate order in an environment with heavy occlusion (e.g., a pile of objects in a bin).

Picking target objects from a pile is especially difficult with a vast variety of shapes and arrangement of deformable objects. Estimating the state of stacked objects is challenging because of 1) Partial observation for the mesh model fitting; 2) Infinite possible patterns of stacking. For example, in Figure 2, the target object (**Dumbbell**) is located on the highest object (**Binder**) and under the obstacle object (**Tennis Ball Holder**). In this scenario, we expect the robot to pick the Tennis Ball Holder first and the Dumbbell afterward, without touching any other objects. However, the detection of only the visible part is not enough to plan such a path to grasp the target object. Typical failure cases occur when the robot tries to pick the wrong object first: 1) Binder because it is the highest object; 2) Dumbbell because it is the target object (Figure 2).

In order to plan the picking order correctly in this situation, it is necessary to understand the scene as “*target object Dumbbell is occluded by obstacle object Tennis Ball Holder*”. This motivates us to introduce *instance occlusion segmentation*, in which the occluded region of each instance (i.e. object) is segmented as well as the visible one. The

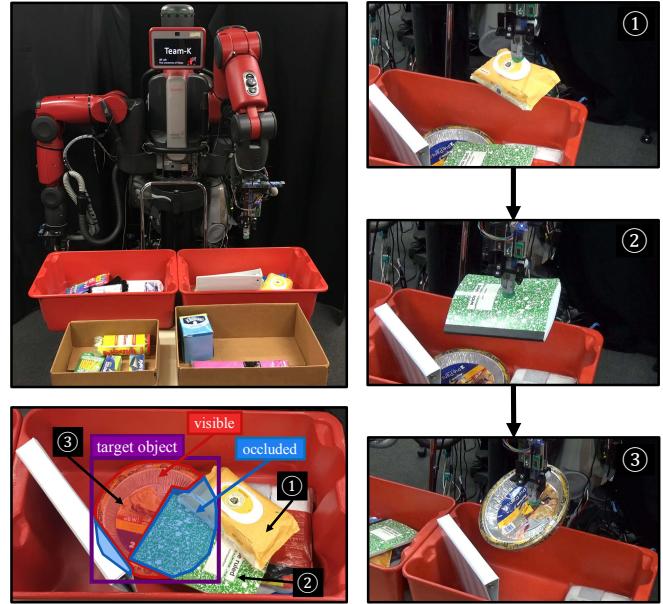


Fig. 1: **Our system** recognizes occlusion status of objects via instance occlusion segmentation, and plans an appropriate picking order to pick the target object from the stacked objects.

segmentation task is aimed at predicting the invisible information (e.g., occluded region) from the visible one using some external knowledge (e.g., by learning from a dataset). This spontaneously leads us to applying a learning-based method as the solution to this problem.

In this paper, we propose a vision system that detects objects and recognizes their visible and occluded region simultaneously. The system is designed to be able to handle novel objects without requiring any task-specific training data (e.g., data collection and human annotation for objects in the bin). To achieve this, our system consists of two components: 1) Image synthesis of a stack of objects solely composed by instance images of these objects, which generates various patterns of stacking and occlusion with ground truth of visible and occluded region masks for each instance; 2) Instance segmentation model with a novel ‘relook’ architecture designed for occlusion segmentation, in which we extend the recent works [7, 8] to recognize and use the density of instances for multi-class segmentation (visible and occluded)

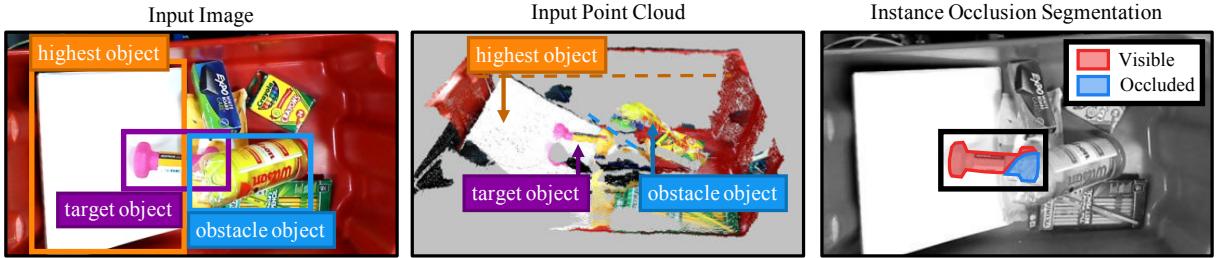


Fig. 2: **Typical case where occlusion understanding is necessary.**

The target object (**Dumbbell**) is located on top of the highest object (**Binder**) and occluded by the obstacle object (**Tennis Ball Holder**). In this case, we expect the robot to pick the Tennis Ball Holder first and the target object afterward. However, the robot mistakenly tries to pick the Binder first in height order picking or the Dumbbell in greedy picking.

of each instance. We also propose a metric for instance occlusion segmentation, which is an extension of common instance segmentation which only segments the visible mask. In the experiments, we provide evaluation results of instance occlusion segmentation using various objects that were used in the Amazon Robotics Challenge (ARC), demonstrating the ability of our system to find and pick the target from a pile of objects.

Our main contributions are:

- Image synthesis framework for learning instance segmentation including occlusion, which is a straightforward extension to recent works [9, 10];
- A novel instance segmentation network model that uses the instance density to segment multi-class masks by extending recent works [7, 8];
- A new metric for instance segmentation of multi-class masks extending recent work of instance segmentation of a single mask [11];
- The integrated system of the above components and demonstration in the picking task from a pile of objects.

## II. RELATED WORKS

### A. Instance Segmentation

Instance segmentation is aimed at predicting the object region mask and its label at the same time. Since instance segmentation is a compound task of bounding box detection and pixel-wise semantic segmentation, previous works propose models that solve the two tasks *sequentially* or *concurrently*. In the *sequential* approach, past works [12]–[14] propose models which propose mask segmentation first and classifies them afterward. On the other hand, *concurrent* segmentation prediction and object detection are recently proposed [7, 8, 15]. These models simultaneously predict object classes, boxes, and masks. In fact, these concurrent approaches prove to be faster and more accurate than sequential ones.

In this paper, we extend the state-of-the-art work [7, 8], for instance occlusion segmentation by seeing it as the multi-class (visible, occluded) extension of the conventional instance segmentation (visible only). Compared to the visible region segmentation, the relationship between nearby objects is more crucial when dealing with occlusion segmentation,

since the occluded region of an instance is caused by other instance’s visible region. This motivates us to extend the previous models in order to learn the connection between predicted object instances. Although previous models concurrently predict visible region masks for each box, there is no connection between them.

Specifically, multi-class extension of *Mask R-CNN* [7] is proposed as *AffordanceNet* [8]. It replaces sigmoid cross entropy loss with softmax to output multi-class region masks, so we refer to the *AffordanceNet* as *Mask R-CNN Softmax* later in this paper.

### B. Image Synthesis for Object Detection

Recently deep learning based methods have improved many machine vision tasks, but since deep learning requires a significant amount of data, it motivates researchers to acquire training data through synthesizing rather than using human annotations. A naive approach of synthesizing training data is using 3D mesh models to render 2D images. Past works use mesh models to learn viewpoint estimation [16], eye gaze direction estimation [17] and object detection [18] for objects in a 2D image. Above work focus on developing ways to make synthetic images closer to real images. On the other hand, it has recently been found that synthesizing only 2D instance images of objects is also effective to train detection models of object bounding boxes [9, 10]. The base idea for this is that if we could generate infinite synthetic images at random and train learning model with it, the model would generalize to real images. Our approach is closer to the latter, and we extend the past works to generate ground truth of object masks (visible, occluded) as well, in addition to the bounding box. Since it is impractical to gather realistic mesh models for various objects, 2D synthesizing is more practical than 3D one. We also show that a small number of instance images is enough to achieve human annotation-level detection performance, while past works [9, 10] use a huge number of instance images for each object.

## III. SYSTEM OVERVIEW

Our proposed system (Figure 3) is composed of two components:

- Instance occlusion segmentation neural networks trained using the generated images (§IV);

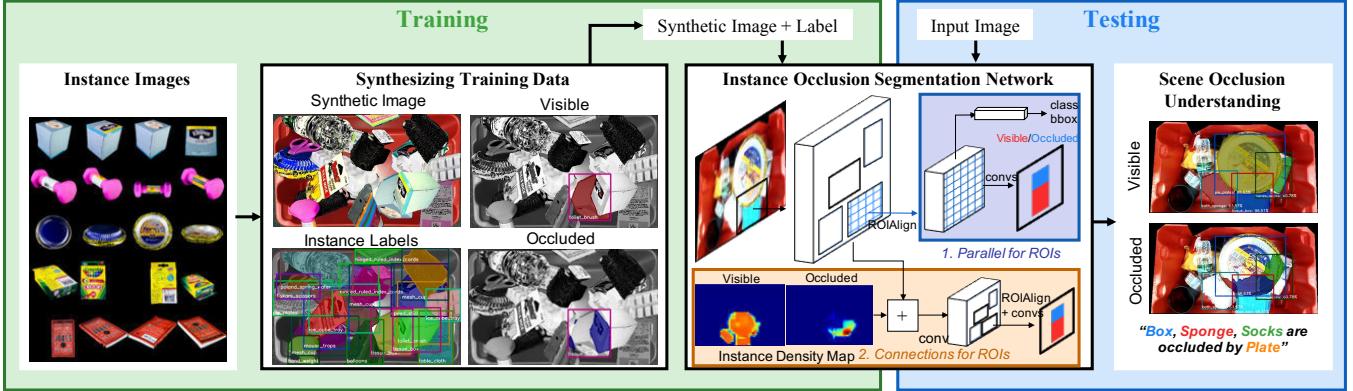


Fig. 3: System overview.

At **train** time, the system receives instance images as input and trains the segmentation model.

At **test** time, the system receives a real image as input and outputs the scene occlusion status for the image.

- 2D image synthesis of a cluttered scene generated from object instance images to handle various objects (§V).

At training time, the system receives instance images of objects as the input to train instance occlusion segmentation model from the generated synthetic images. At testing time, a real image is the input and the occlusion status of the scene is predicted and outputted. Since our proposed system only requires instance images of objects of interest, we can quickly gather them from web or with a standard camera. This is important for enabling the vision system to handle various objects without human labeling, which is especially hard for instance occlusion segmentation, and for the applications to warehouse picking in e-commerce services for example.

The network output is the set of visible and occluded masks of each object instance and the occlusion status. The hierarchy of stacking like “*Plate is on Box*“ is interpreted automatically by scanning each pixel to see what object is visible and what objects are occluded at the pixel.

#### IV. INSTANCE OCCLUSION SEGMENTATION NEURAL NETWORKS

##### A. Network Architecture

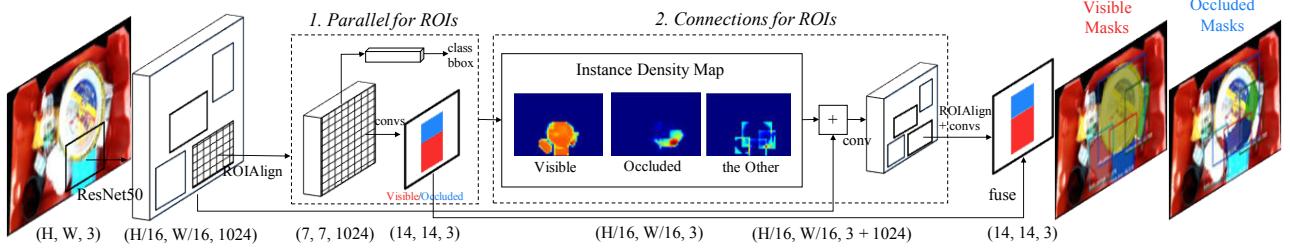
1) *Mask R-CNN*: We begin by reviewing the network architecture of Mask R-CNN [7]. Mask R-CNN is an extension of Faster R-CNN [19], which was previously proposed as a model for bounding box based object detection. To extend the model for instance segmentation, another branch of predicting instance mask was introduced and added to the existing class and bounding box prediction branches. In the mask branch, the instance mask is predicted without predicting object class by relying on the classification of the classification branch. This is achieved by predicting the class number of instance masks in the mask branch, extracting the mask of predicted class by classification branch afterward. Each branch of the object class, box, and mask is predicted in parallel after extracting features by ROIAlign (an extension of ROIPooling [19]) at the Region of Interest (ROI) proposed by the preceding Region Proposal Networks (RPN) [19].

Mask R-CNN is fast because the prediction of each instance information is made in *parallel* and *independently*

using the features extracted by ROIAlign for each proposed ROI. However, since the prediction about each instance after ROI feature extraction is independent of other ROIs/instances, this model is not suited for learning the relationship among instances. Therefore, although effective for instance segmentation of the visible region, in which the relationship among instances is not so important, Mask R-CNN is not expected to be suitable for instance occlusion segmentation, in which the relationship between masks is crucial for correctly inferring the occlusion state of the instances. This leads us to introduce the inter-instance connection, which is described in §IV-A.2.

2) *Relook Architecture: Inter-Instance Connection*: In order to learn the relationship and dependency between instances, it is necessary to have connections among the representations of each instance in the neural network. To do this, we convert the instance masks predicted in the first stage (left in Figure 4) to a density map (middle in Figure 4) and use it to predict instance masks in the second stage. The two instance masks of the first and second stage are added in pixel-wise (fused) as the final result, being segmentation loss computed for this fused result. The first stage is Mask R-CNN (softmax) (§II-A). The second stage can be interpreted as a “*relook*” architecture for learning and predicting inter-instance connections, so we call this model Mask R-CNN (relook).

The final layer predicts three masks: visible, occluded and other (not part of the object), being the density map also generated for each of the three. For learning inter-instance connection, we concatenate the density map with the features extracted by the feature extractor. We use ResNet50-C4 and ResNet101-C4 (fourth stage output of ResNetX [20]) as in Mask R-CNN [7], and apply a convolutional layer. After that, ROIAlign, ‘res5’ (5th layer of ResNetX) and a deconvolutional layer is applied with sharing parameters with the first stage. On top of that, the final convolutional layer is applied to predict three masks for each instance. We use ReLU [20] as the activation function for the hidden layers, and the kernel size of the convolutional layer is 3 for the hidden layer and 1 for the output layer.



**Fig. 4: Instance occlusion segmentation network with inter-instance connection.**

The network architecture can be interpreted as a pipeline of two components:

- 1) Mask R-CNN (softmax) for per-instance prediction;
- 2) Instance density map for inter-instance connection.  
The tensor shape of the layer output is shown in the bottom with (height, width, channel).

## B. Implementation Detail

At most, we followed the implementation of the original paper Mask R-CNN [7] and Faster R-CNN [19]. For input image size, we set 600 as the minimum axis size and 1000 as the maximum axis size following [19]. We trained RPN using three ratios and four anchor scales with no threshold of ROI proposal size, which was set to 16 in [19]. We use 512 as the hidden channel size of RPN, value which was set to 1024 in [7], since we noticed it was too large for small datasets.

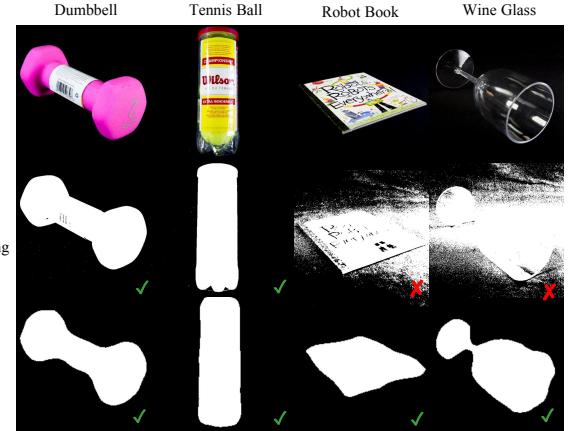
At the training time of Mask R-CNN, 512 ROI proposals, which have *foreground : background* = 1 : 4 ratio, are used to train classification and box regression branch, and the foreground ROIs are used to train the mask branch. At the testing time, it first regresses class and bounding box of the instances and applies non-maximum suppression (NMS), then using the fewer and more accurate bounding boxes for ROI feature extraction by ROIAlign with following mask prediction. This causes a little difference in the prediction process between training and testing; however, it achieves mask prediction using the more accurate bounding box to get better results. This prediction difference may also cause bad effects because of the difference of the density map: at training time NMS is not applied, so the density map appears to have more instances than that of testing time. However, a little surprisingly, this prediction difference did not cause adverse effects, but rather we had better results with this prediction process than using the same prediction process at both training and testing time.

## V. IMAGE SYNTHESIS FOR LEARNING INSTANCE OCCLUSION SEGMENTATION

### A. Instance Image Gathering

We use instance images distributed at the Amazon Robotics Challenge as *ItemData* for both known and novel objects which will appear during the task. There are few images (4–6) taken from different viewpoints for each object with black background, as shown in the top images of Figure 5.

Although the background of the instance image is static, we found that it is difficult to extract the foreground mask for each instance through pixel value thresholding. This is especially true for objects with various colors (e.g., Robot



**Fig. 5: Foreground mask extraction for instance image.**  
Sometimes foreground mask extraction through pixel value threshold fails even for images with simple background (middle images). In this work we counteract this problem by training a small convolutional neural network model (bottom images) [21].

Book) or transparent (e.g., Wine Glass), as shown in the middle of Figure 5.

A similar problem is also pointed out in previous work [10]: they had objects for which it is difficult to extract foreground mask by thresholding of the depth image, in particular with transparent objects such as Cola Bottle. To overcome this difficulty, they trained a small convolutional neural network (ConvNet) model [21] using the mask acquired from thresholding of the depth image as the ground truth. We applied the same approach, using the mask acquired from pixel value thresholding using instance images of 112 objects. The ground truth foreground mask is not perfect because it is not annotated by a human; however, the segmentation result in Figure 5 shows that the network successfully generalized the foreground segmentation. In this case, the automatically generated mask is usually bigger than the perfect mask, while in [10] the generated mask from depth image is usually smaller than the perfect one.

### B. Blending

Blending is necessary for 2D image synthesis to remove boundary artifacts when we put the instance images onto

the background image. We apply gaussian blurring following [10] with a random sigma which ranges from 0 to 1.

### C. Data Augmentation

Data augmentation is crucial for generating synthetic data to train detection models that will generalize to real images, especially when we only have a few instance images (4–6), compared to 600 images found in [10]. We applied color data augmentation in addition to the geometric augmentation present in past work [10]. To make the system more robust to changes in brightness and light reflection, we also applied multiplication to S and V channels after converting the RGB image into HSV color space with random selection of scale between 0.5 to 2.0. For geometric augmentation, we applied affine transformation with a random scale in the range from 0.5 to 1.0, translation from -16 to 16 pixels, rotation from -180 to 180 degrees, and shear from -16 to 16 degrees, which we believe is similar to the augmentation of 2D/3D rotation described in [10]. Figure 6 shows some examples of the augmented result.



Fig. 6: Data augmentation example.

### D. Image Synthesis of a Pile of Objects with Ground Truth

We generate the ground truth labels and masks in addition to the synthetic image of stacked objects. Figure 7 shows an example of the synthesized image and visualization of the ground truth. While stacking the instance images onto the background image, we apply image blending (§V-B) and data augmentation (§V-C) for each instance image at random. The foreground mask of each instance image is acquired by ConvNet (§V-A), and since the filled region of already stacked instances is known, we can get masks of both visible and occluded regions (Figure 7c and 7d).

## VI. A METRIC FOR INSTANCE OCCLUSION SEGMENTATION

The instance segmentation model of multiple masks (affordance masks) in [8] was evaluated as a pixel-wise segmentation task to compare with other state-of-the-art semantic segmentation models. This was possible because their model segments only the visible region of each instance and there was no overlap among the masks. Also, their evaluation ignores the accuracy of the predicted number of instances: even if 3 instances are predicted for an image with 2 instances in ground truth, there is no penalty for the over-counting

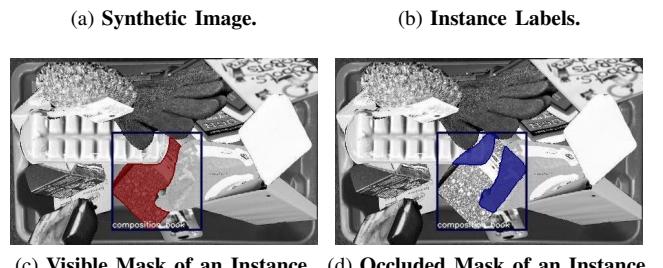
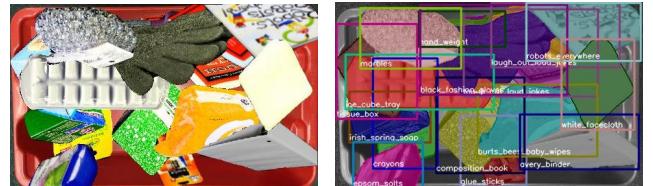


Fig. 7: Image synthesis example.

1 instance. This happens because the predicted bounding boxes and affordance masks will be converted to an image of pixel-wise affordance label to evaluate it in the semantic segmentation manner. In order to attain the goal of picking targets from a clutter, however, detecting the correct number of instances is also necessary, as well as the segmentation of visible and occluded masks. This motivated us to find another metric to evaluate instance occlusion segmentation.

Recently, the metric Panoptic Quality ( $PQ$ ) is proposed to evaluate the accuracy of both detection and segmentation in instance segmentation [11].  $PQ$  is represented as the multiplication of Detection Quality ( $DQ$ ) and Segmentation Quality ( $SQ$ ):

$$DQ = \frac{|TP|}{|TP| + \frac{1}{2}|FP| + \frac{1}{2}|FN|} \quad (1)$$

$$SQ = \frac{\sum_{(p,g) \in TP} IoU(p, g)}{|TP|} \quad (2)$$

$$PQ = DQ \cdot SQ, \quad (3)$$

where  $TP$ ,  $FP$  and  $FN$  represent the set of true positive, false positive and false negative instances, and  $p, g$  represent predicted and ground truth instances, respectively.  $IoU$  is the intersect over union of predicted and ground truth masks for a single mask-class:

$$IoU(p, g) = \frac{|p \cap g|}{|p \cup g|} \quad (4)$$

where  $p$  and  $g$  are the predicted and ground truth masks for a single mask-class.  $PQ$  is computed for each object class, and the  $PQs$  for all classes are averaged as “means of  $PQ$ ” ( $mPQ$ ). For computation of  $DQ$ , we use the visible mask of predicted and ground truth to find the matched instances with  $IoU$  threshold of 0.5 between these predicted masks.

Since the  $SQ$  was proposed as the metric of instance segmentation, which is a single mask segmentation for each object, we extended the metric to be able to evaluate multi

masks segmentation for each instance as  $SQ_{multi}$ :

$$SQ_{multi} = \frac{\sum_{(p,g) \in TP} mIoU(p, q)}{|TP|} \quad (5)$$

$$\text{where } mIoU(p, q) = \sum_{m \in M} IoU_m(p, q) / |M|. \quad (6)$$

$M$  represents the set of possible instance masks, of which we have three in the instance occlusion segmentation: background, visible and occluded.  $mIoU$  (mean of  $IoU$ ) is the averaged  $IoU$  over the mask classes  $m \in M$ . The  $IoU$  and  $mIoU$  have been used in the previous works of semantic segmentation [22, 23] to evaluate the accuracy of predicted masks comparing it with the ground truth, so we believe the  $PQ_{multi} = DQ \cdot SQ_{multi}$  is an appropriate metric for instance occlusion segmentation. In the following we address to  $PQ_{multi}$  as  $PQ$  for simplicity.

## VII. EXPERIMENTS

#### A. Instance Occlusion Segmentation of ARC2017 Objects

*1) Objects for Evaluation:* We evaluate our system with the 40 objects used in the Amazon Robotics Challenge 2017 (ARC2017) shown in Figure 8, each of which had 4-6 instance images distributed at the competition. We believe these objects have a broad diversity and are sufficient to demonstrate picking from a pile of objects.

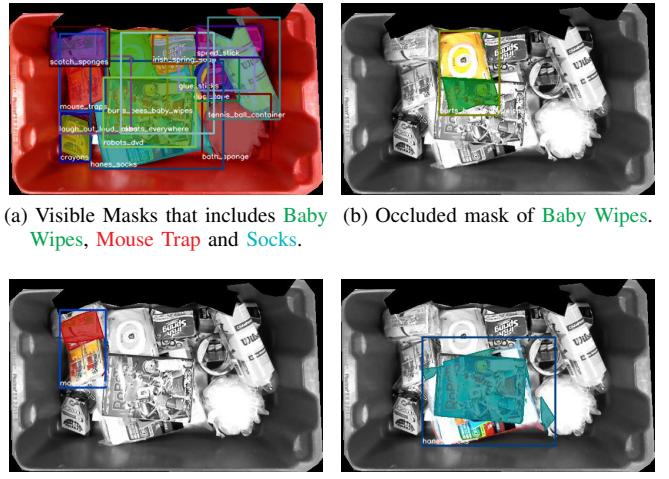
In the following experiments, we use the instance images distributed at ARC2017 to evaluate our image synthesis framework (§V) comparing with human annotated data we created using the real objects. The instance occlusion segmentation model (§IV) is trained with the human-annotated or synthetic training dataset and is evaluated with the human-annotated testing dataset.



Fig. 8: **40 objects** used at ARC2017.

**2) Human-annotated Dataset for Evaluation:** For the evaluation of both the model and image synthesis, we created a dataset of instance occlusion segmentation shown in Figure 9. Since annotating an occluded region is challenging we created a set of sequential camera frames in which a pile of objects is cleared from the top. With annotating the visible mask of objects in all frames of a video captured from a fixed camera, the visible masks are backtracked to

acquire the occluded masks. We created 21 videos (split into  $train : test = 14 : 7$ ) in which the 40 objects appear 7 times each (3-5 times in the train split).



**Fig. 9: Annotations of instance occlusion segmentation.**  
 Occluded mask of objects is visualized separately. Note that its color (**Baby Wipes**, **Mouse Trap**, **Socks**) corresponds to the mask of visualization of visible masks.

**3) Model Evaluation: Softmax vs. Relook:** We evaluated the proposed relook architecture with a human-annotated dataset for both training and testing, with the comparison to the softmax extension of Mask R-CNN [8]. Table II shows the quantitative results, in which “Softmax” denotes Mask R-CNN Softmax, “Softmax\_x2” denotes Softmax whose mask loss is scaled by 2, and “Relook” denotes Mask R-CNN with the relook architecture. The reason why we added the experiment of “Softmax\_x2” is because our proposed model has two mask losses at first and second stage, possibly making it unfair to compare it with “Softmax”, which has no scaling factor to the mask loss. Since the learning result of RPN has lots of noise caused by randomness, we show results averaged in 10 - 15 times experiments for each model. For the fair comparison, we use ResNet50 feature extractor using learning rate 0.0375 with 3 GPUs in which the learning rate is scaled by the number of GPUs in following experiments without note:  $0.00375 = 3 \times 0.00125$ , as proposed in [24].

The results of  $mPQ$  in Table II show that the proposed relook architecture surpasses the existing models and is effective in performing instance occlusion segmentation. We also show results of  $mAP$  (mean averaged precision) that was used as the metric of instance segmentation of visible mask in the VOC [22] and COCO [23] competitions. The results of the  $mAP$  show that our model surpasses previous models in both instance segmentation of visible masks and detection quality.

We also trained the network with a different backbone, ResNet101. Its qualitative results are shown in Table I.

*4) Dataset Evaluation: Human-annotated vs. Synthetic:* We evaluated our image synthesis framework by training the proposed model with either synthetic or human-annotated

TABLE I: Instance occlusion segmentation on test data with ResNet101 backbone.

Scene	Ground Truth		Predicted		Detected Instance
	Visible	Occluded	Visible	Occluded	
					Binder ✓ Tennis Ball ✓ Green Book ✓ Table Cloth ✓
					Tape ✓ Table Cloth ✓ Robot Book ✓ Pie Plate ✓
					Baby Wipes ✓ Salt ✓ Crayons ✓ Green Book ✓ Mesh Cup ✗

TABLE II: Softmax vs. Relook.

Model	mPQ	mSQ	mDQ	mAP
Softmax [8]	13.4	24.7	40.7	46.1
Softmax_x2 [8]	13.8	25.2	41.9	47.1
Relook (Ours)	<b>14.4</b>	<b>26.0</b>	<b>42.8</b>	<b>48.6</b>

data. After training, the model performance is evaluated using the test split of the human-annotated dataset. Table III shows the averaged results of 3 experiments using the image synthesis referring results in §VII-A.3. It shows that our image synthesis using 4 - 6 instance images (14.2) is as effective as the result using a small human annotated dataset (14.4) for learning instance occlusion segmentation.

TABLE III: Human-annotated vs. Synthetic.

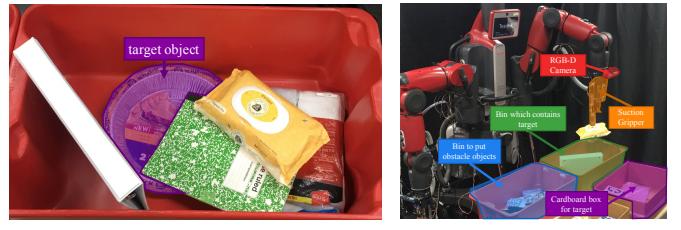
Model	Dataset	mPQ
Softmax [8]	Annotated	13.4
	Synthetic	13.5
Relook (ours)	Annotated	14.4
	Synthetic	<b>14.2</b>

### B. Application to Warehouse Picking

As an application of our system, we demonstrate the picking task of a target object from a pile of objects, as shown in Figure 10a. Figure 10b shows the workspace configuration: there is a bin which contains the target object, a bin to place obstacle objects into it, and a cardboard box for the robot to place the target object. For this demonstration, we have extended the picking task we developed in the previous works [5, 25]. Figure 11 shows the sequential frames of recognition result (Figure 11a-11c), workspace overview (Figure 11d-11i) and emphasis on the robot hand (Figure 11j-11o). The recognition result shows the input image, predicted visible and occluded masks, and the target object based on the occlusion understanding.

In this experiment, we set the threshold of occlusion ratio to 0.3; the occlusion ratio is the ratio of occluded pixels compared with the total pixels of an instance. We set the threshold of inter-instance occlusion ratio to 0.1 to judge that the instance is occluded by the other. For the model of

this demonstration, we use ReNet101 feature extractor as the backbone of our proposed model to have a better recognition accuracy. The successful clearing of obstacle objects and picking targets in the demonstration shows that our proposed system is effective and applicable in the real-world picking task.



(a) The Scene of Stacked Objects. (b) Workspace.

Fig. 10: Picking Demo Configuration.

## VIII. CONCLUSIONS

We presented a vision system that only requires a few instance images of objects to learn instance occlusion segmentation. The system consists of 1) Image synthesis with ground truth of occluded region mask of each instance; 2) Instance segmentation networks that learn inter-instance relationship, which is essential for the segmentation of occluded regions. We evaluated the proposed image synthesis and segmentation model via the ablation studies and presented the effectiveness of the proposed system in the real picking task from a pile of objects.

## ACKNOWLEDGEMENT

We thank Shun Hasegawa and Yuto Uchimi for contributions to the software and hardware development of the picking system for demonstration with real-world robot experiments.

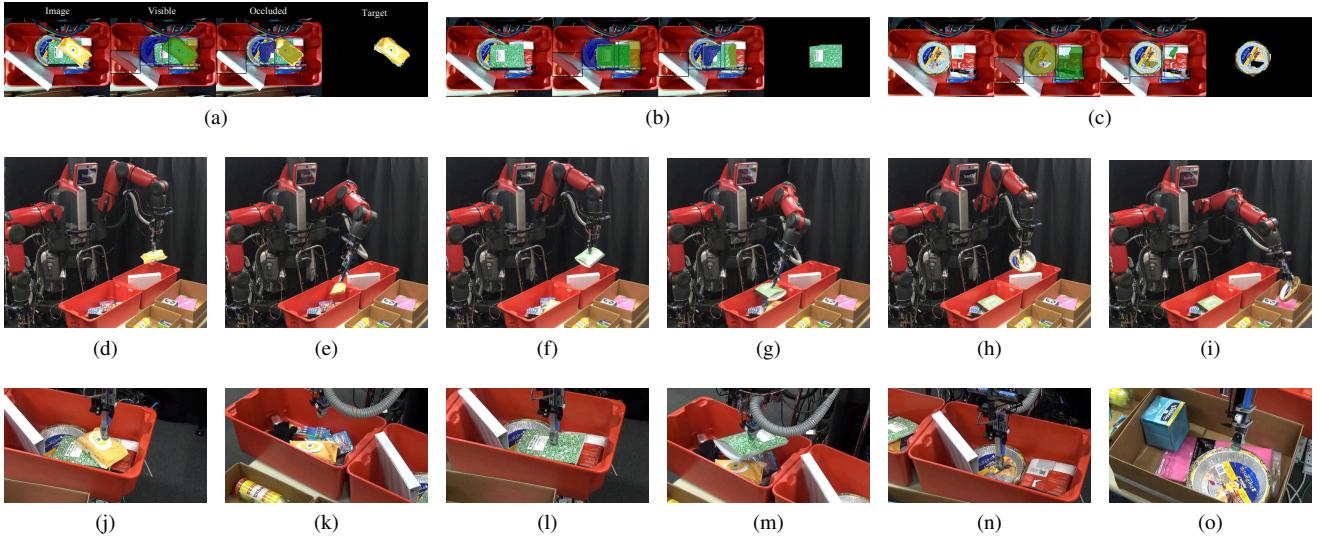


Fig. 11: Picking Task Demonstration of a Target from a Pile of Objects.

## REFERENCES

- [1] Rico Jonschkowski, Clemens Eppner, Sebastian Höfer, Roberto Martín-Martín, and Oliver Brock. Probabilistic multi-class segmentation for the Amazon picking challenge. *IEEE International Conference on Intelligent Robots and Systems*, No. i, pp. 1–7, 2016.
- [2] Carlos Hernandez, Mukunda Bharatheesha, Wilson Ko, Hans Gaiser, Jethro Tan, Kanter van Deurzen, Maarten de Vries, Bas Van Mil, Jeff van Egmond, Ruben Burger, Mihai Morariu, Jihong Ju, Xander Germann, Ronald Ensing, Jan Van Frankenhuyzen, and Martijn Wisse. Team delft’s robot winner of the amazon picking challenge 2016. *Lecture Notes in Computer Science*, Vol. 9776 LNAI, pp. 613–624, 2017.
- [3] Max Schwarz, Anton Milan, Christian Lenz, Aura Munoz, Arul Selvam Periyasamy, Michael Schreiber, Sebastian Schuller, and Sven Behnke. NimbRo picking: Versatile part handling for warehouse automation. *Proceedings - IEEE International Conference on Robotics and Automation*, No. May, pp. 3032–3039, 2017.
- [4] Andy Zeng, Kuan Ting Yu, Shuran Song, Daniel Suo, Ed Walker, Alberto Rodriguez, and Jianxiong Xiao. Multi-view self-supervised deep learning for 6D pose estimation in the Amazon Picking Challenge. *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 1386–1393, 2017.
- [5] Kentaro Wada, Kei Okada, and Masayuki Inaba. Probabilistic 3d multilabel real-time mapping for multi-object manipulation. In *Proceedings of the International Conference on Robotics and Systems*, 2017.
- [6] Andy Zeng, Shuran Song, Kuan-Ting Yu, Elliott Donlon, François Hogan, Maria Bauza, Daolin Ma, Orion Taylor, Melody Liu, Eudald Romo, Nima Fazeli, Ferran Alet, Nikhil Chavan Dafle, Rachel Holladay, Isabella Morona, Prem Qu Nair, Druck Green, Ian Taylor, Weber Liu, Thomas Funkhouser, and Alberto Rodriguez. Robotic Pick-and-Place of Novel Objects in Clutter with Multi-Affordance Grasping and Cross-Domain Image Matching. *IEEE International Conference on Robotics and Automation (ICRA)*, p. In review., 2018.
- [7] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask R-CNN. 2017.
- [8] Thanh-Toan Do, Anh Nguyen, Ian Reid, Darwin G Caldwell, and Nikos G Tsagarakis. AffordanceNet: An End-to-End Deep Learning Approach for Object Affordance Detection. 2017.
- [9] Georgios Georgakis, Arsalan Mousavian, Alexander C Berg, and Jana Kosecka. Synthesizing Training Data for Object Detection in Indoor Scenes. 2017.
- [10] Debidatta Dwibedi, Ishan Misra, and Martial Hebert. Cut, Paste and Learn: Surprisingly Easy Synthesis for Instance Detection. 2017.
- [11] Alexander Kirillov, Kaiming He, Ross Girshick, Carsten Rother, and Piotr Dollár. Panoptic Segmentation. pp. 1–9, 2018.
- [12] Pedro O. Pinheiro, Ronan Collobert, and Piotr Dollar. Learning to Segment Object Candidates. pp. 1–10, 2015.
- [13] Pedro O. Pinheiro, Tsung Yi Lin, Ronan Collobert, and Piotr Dollár. Learning to refine object segments. *Lecture Notes in Computer Science*, Vol. 9905 LNCS, pp. 75–91, 2016.
- [14] Jifeng Dai, Kaiming He, Yi Li, Shaoqing Ren, and Jian Sun. Instance-sensitive Fully Convolutional Networks. pp. 1–15, 2016.
- [15] Yi Li, Haozhi Qi, Jifeng Dai, Xiangyang Ji, and Yichen Wei. Fully Convolutional Instance-aware Semantic Segmentation. 2016.
- [16] Hao Su, Charles R. Qi, Yangyan Li, and Leonidas J. Guibas. Render for CNN: Viewpoint estimation in images using CNNs trained with rendered 3D model views. *Proceedings of the IEEE International Conference on Computer Vision*, Vol. 2015 International Conference on Computer Vision, ICCV 2015, No. Sec 2, pp. 2686–2694, 2015.
- [17] Ashish Shrivastava, Tomas Pfister, Oncel Tuzel, Josh Susskind, Wenda Wang, and Russ Webb. Learning from Simulated and Unsupervised Images through Adversarial Training. 2016.
- [18] Stefan Hinterstoisser, Vincent Lepetit, Paul Wohlhart, and Kurt Konolige. On Pre-Trained Image Features and Synthetic Images for Deep Learning. 2017.
- [19] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 39, No. 6, pp. 1137–1149, 2017.
- [20] Vinod Nair and Geoffrey E Hinton. Rectified Linear Units Improve Restricted Boltzmann Machines. *Proceedings of the 27th International Conference on Machine Learning*, No. 3, pp. 807–814, 2010.
- [21] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Vol. 07-12-June, pp. 3431–3440, 2015.
- [22] Mark Everingham, S. M Ali Eslami, Luc Van Gool, Christopher K I Williams, John Winn, and Andrew Zisserman. The Pascal Visual Object Classes Challenge: A Retrospective. *International Journal of Computer Vision*, Vol. 111, No. 1, pp. 98–136, 2014.
- [23] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C Lawrence Zitnick, and Piotr Dollár. Microsoft COCO: Common Objects in Context. 2014.
- [24] Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Apo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. Accurate, Large Minibatch SGD: Training ImageNet in 1 Hour. 2017.
- [25] Shun Hasegawa, Kentaro Wada, Yusuke Niitani, Kei Okada, and Masayuki Inaba. A Three-Fingered Hand with a Suction Gripping System for Picking Various Objects in Cluttered Narrow Space. pp. 1164–1171, 2017.