



저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.



변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)

Master's Thesis

Vision based Generous Deep Neural Network  
Grasping Detector for Robot Grasping Task

Yong-Hyeok Seo

Department of Electrical Engineering

Ulsan National Institute of Science and Technology

2021

# Vision based Generous Deep Neural Network Grasping Detector for Robot Grasping Task

Yong-Hyeok Seo

Department of Electrical Engineering

Ulsan National Institute of Science and Technology

# Vision based Generous Deep Neural Network Grasping Detector for Robot Grasping Task

A thesis/dissertation submitted to  
Ulsan National Institute of Science and Technology  
in partial fulfillment of the  
requirements for the degree of  
Master of Science

Yong-Hyeok Seo

12/15/2020 of submission

Approved by



---

Advisor

Se-Young Chun

# Vision based Generous Deep Neural Network Grasping Detector for Robot Grasping Task

Yong-Hyeok Seo

This certifies that the thesis/dissertation of Yong-Hyeok Seo is approved.


12/15/2020



Advisor: Se-Young Chun



Sung-Phil Kim: Thesis Committee Member #1



Jeong Hwan Jeon: Thesis Committee Member

#2

## Abstract

Rotation invariance has been an important topic in computer vision tasks such as face detection [1], texture classification [2] and character recognition [3], to name a few. The importance of rotation invariant properties for computer vision methods still remains for recent DNN based approaches. In general, DNNs often require a lot more parameters with data augmentation with rotations to yield rotational-invariant outputs. Max pooling helps alleviating this issue, but since it is usually  $2 \times 2$  [4], it is only for images rotated with very small angles. Recently, there have been some works on rotation-invariant neural network such as rotating weights [5, 6], enlarged receptive field using dialed convolutional neural network (CNN) [7] or a pyramid pooling layer [8], rotation region proposals for recognizing arbitrarily placed texts [9] and polar transform network to extract rotation-invariant features [10].

Applications of deep neural network based object and grasp detections could be expanded, significantly when the network output is processed by a high-level reasoning over relationship of objects. Recently, robotic grasp detection and object detection with reasoning have been investigated using deep neural networks (DNNs). There have been effects to combine these multi-tasks using separate networks so that robots can deal with situations of grasping specific target objects in the cluttered, stacked, complex piles of novel objects from a single RGB-D camera. We propose a single multi-task DNN that yields an accurate detections of objects, grasp position and relationship reasoning among objects. Our proposed methods yield state-of-the-art performance with the accuracy of 98.6% and 74.2% with the computation speed of 33 and 62 frame per second on VMRD and Cornell datasets, respectively. Our methods also yielded 95.3% grasp success rate for novel object grasping tasks with a 4-axis robot arm and 86.7% grasp success rate in cluttered novel objects with a humanoid robot

## Contents

<b>Contents</b>	<b>ii</b>
<b>List of Figures</b>	<b>iv</b>
<b>List of Tables</b>	<b>vi</b>
<b>I. Introduction</b>	<b>1</b>
<b>II. Real-time Highly Accurate Grasping Detection with Rotation Ensemble Module</b>	<b>4</b>
2.1 Related works . . . . .	4
2.2 Grasping Detection . . . . .	6
2.3 Rotation Ensemble Module . . . . .	7
2.3.1 Parameter descriptions of the proposed OSD method . . . . .	7
2.3.2 Rotation ensemble module (REM) . . . . .	7
2.4 Loss functions . . . . .	9
2.5 Simulations and Experiments . . . . .	10
2.5.1 Implementation details . . . . .	10
2.5.2 Benchmark dataset and novel objects . . . . .	11
2.5.3 Results for in-house implementations of previous works . . . . .	12
2.5.4 Results for our proposed REM on the Cornell dataset . . . . .	13
<b>III. A Single Neural Network for Multi-Taskf</b>	<b>15</b>
3.1 Related Works . . . . .	15
3.2 Multi-task robot grasping . . . . .	16
3.2.1 Single object robot grasping . . . . .	16
3.2.2 Multi-task robot grasping . . . . .	16
3.3 Reparametrization of 15D representation . . . . .	16
3.3.1 Anchor box: w, h in each cell . . . . .	17
3.3.2 Anchor box: orientation in each cell. . . . .	17
3.3.3 Object class: cls in each cell. . . . .	17

---

3.3.4	FC and CC in each cell. . . . .	18
3.4	Proposed FCNN with predictions across scales . . . . .	18
3.5	Reasoning post-processing: from class to index . . . . .	20
3.6	Loss functions . . . . .	20
3.7	Experimental Evaluations . . . . .	21
3.7.1	Implementation details . . . . .	21
3.7.2	Evaluations on VMRD and Cornell datasets . . . . .	21
3.8	Results . . . . .	22
3.8.1	Simulation results on VMRD dataset . . . . .	22
3.8.2	Simulation results on Cornell dataset . . . . .	23
<b>IV.</b>	<b>Toward Robot Demonstration in Real-environment</b>	<b>26</b>
4.1	Evaluation of multi-tasks OD, GD, reasoning with Baxter . . . . .	26
4.2	Evaluation of GD with 4-axis robot arm . . . . .	26
4.3	Results of Robot Evaluation in Real-environment . . . . .	27
4.3.1	Results of multi-task OD, GD, reasoning with Baxter . . . . .	27
4.3.2	Results of GD with 4-axis robot arm . . . . .	28
4.4	Discussion . . . . .	28
<b>V.</b>	<b>Conclusion</b>	<b>30</b>
	<b>References</b>	<b>31</b>



## List of Figures

1.1	REM module performance summary of computation time (frame per second) vs. grasp detection accuracy on the Cornell dataset with object-wise data split. . . . .	3
1.2	(top left panel) GD with grasp candidates (black rectangles) and the best grasp (green and red rectangle) (right panels) multi-tasks of GD, OD and relationship reasoning. (bottom left panel) computation speed (FPS) vs. prediction accuracy (mAP) for multi-task grasping detection of our method achieving the state-of-the-art performance in both accuracy and speed and other previous work of (a) [11], (b,c) [12]. . . . .	3
2.1	(a) A 5D detection representation with location $(x, y)$ , rotation $\theta$ , gripper opening with $w$ and plate size $h$ . (b) For a (2,2) grid cell, all parameters for 5D representation are illustrated including a pre-defined anchor box (black dotted box) and a 5D detection representation (red box). . . . .	6
2.2	An illustration of incorporating our proposed REM in a DNN for robot grasp detection (a) and the architecture of our proposed REM with rotation convolutions (b). . . . .	8
2.3	Images from the Cornell dataset . . . . .	11
2.4	Grasp detection accuracy over epoch on the Cornell dataset using various methods for angle predictions: Rot: rotation anchor box, Cls: classification, Reg: regression, REM: ours. . . . .	12
2.5	Grasp detection results on the Cornell dataset for (a) Reg, a modern version of Redmon [13], (b) Cls, a modern version of Guo [14], (b) Rot, a modern version of Zhou [15] and (d) our proposed Cls+REM. (e) Ground truth labels in Cornell dataset. Black boxes are grasp candidates and green-red boxes are the best grasp among them. . . . .	14
3.1	Proposed FCNN architecture based on Darknet. . . . .	18
3.2	Schematic pipelines of Zhang [11, 12] vs ours. Mark 'N' means neural network and 'P' means post-processing. . . . .	19
3.3	VMRD dataset. . . . .	22

**LIST OF FIGURES**

---

3.4	Multi-task detection results for VMRD. The 1 <sup>st</sup> row is GT and the 2 <sup>nd</sup> row is the results of our proposed methods. Note that our method yielded correct reasoning result for “Stapler” while GT incorrectly describes it. . . . .	23
3.5	GD results on Cornell dataset using our methods without and with predictions across scales. . . . .	24
4.1	(a) our real multi-task evaluation environment (Baxter). (b) our robot grasping experiment with 4-axis robot. . . . .	27
4.2	Target grasp detection results in (a) cluttered scene, (b) stacking scene and (c) challenging invisible scene. . . . .	29

## List of Tables

2.1	Ablation studies on the Cornell dataset for anchor box of $w, h$ with various ratios or one ratio and angle prediction methods with Reg, Cls, Rot. . . . .	12
2.2	The ablation studies on the Cornell dataset for our REM with RC, RA and RL. . . . .	13
2.3	Performance summary on Cornell dataset. Our proposed method yielded state-of-the-art prediction accuracy in both image-wise (Img) and object-wise (Obj) splits with real-time computation. The unit for performance is % . . . . .	13
3.1	Self-evaluation summary on VMRD. . . . .	22
3.2	Performance summary on VMRD dataset. . . . .	23
3.3	Summary on Cornell data (25% IOU). . . . .	25
4.1	Performance summary of grasping tasks for cluttered (CS), stacking (SS) and invisible (IS) scenes. . . . .	27

---

# Introduction

---

Robot grasping of novel objects has been investigated extensively, but it is still a challenging open problem in robotics. Humans instantly identify multiple grasps of novel objects (perception), plan how to pick them up (planning) and actually grasp it reliably (control). However, accurate robotic grasp detection, trajectory planning and reliable execution are quite challenging for robots. As the first step, detecting robotic grasps accurately and quickly from imaging sensors is an important task for successful robotic grasping.

In this paper, we propose a rotation ensemble module (REM) for robotic grasp detection using convolutions that rotates network weights. This special structure allows the DNN to select rotation convolutions for each grid. Our proposed REM were evaluated for two different tasks: robotic grasp detection on the Cornell dataset [16, 17] and real robotic grasping tasks with novel objects that were not used during training. Our proposed REM was able to outperform state-of-the-art methods such as [15] by achieving up to 99.2% (image-wise), 98.6% (object-wise) accuracy on the Cornell dataset as shown in Fig. 1.2 with  $5\times$  faster computation than [15]. Our proposed method was also able to yield up to 93.8% success rate for the real-time robotic grasping task with a 4-axis robot arm for novel objects and to yield reliable grasps for multiple objects unlike rotation anchor box.

Robot grasping of particular target objects in cluttered, stacked and complex piles of novel objects is a challenging open problem. Humans instantly identify/locate target objects and their nearby objects (object detection or OD), figure out location-wise relationship among objects (reasoning), and detect multiple grasps of the targets and their associated objects (grasp detection or GD). However, these tasks

are still quite challenging for robots. Locating the targets and nearby objects in the piles of objects, reasoning their relationships and detecting multiple robotic grasps accurately and quickly are important multi-tasks for successful robotic grasping.

Deep learning based approaches have been actively investigated for robot grasp detection since the work of Lenz *et al.* [16, 17]. Thanks to the Cornell robotic grasp detection open database [16] and the advance of deep learning techniques, many approaches have been proposed [13, 14, 18]. The current state-of-the-art GD accuracy on the Cornell dataset is up to 97.7% [15]. The Cornell dataset contains images with a single object and multiple grasp labels. Deep neural networks (DNNs) trained with this dataset generally yielded multiple grasps for a single object or multiple objects that are separately placed. Due to the capacity of DNNs, they often yielded good GD results for novel objects. However, the Cornell dataset lacks of robotic grasping in the cluttered piles of objects. Moreover, it is especially challenging which object the robot has to grasp first in order not to damage other objects for cluttered or stacked objects or in order to efficiently grasp specific target objects. In other words, a robot must know if an object is on another object in the piles of objects for successful grasping.

Recently, Zhang *et al.* proposed multi-task convolution robotic grasping networks to address the problem of combining GD and OD with relationship reasoning in the piles of objects [11]. This method consists of multiple DNNs that are responsible for generating local feature maps, GD, OD and relationship reasoning separately. More specifically, features are extracted using ResNet-101 based region proposal network (RPN) and then are fed into three DNNs corresponding to three tasks: OD, GD and relationship prediction among objects to perform grasping considering relationships and orderings (relationship reasoning). This approach facilitates matching for reasoning and achieved high GD accuracy of 70.1% on the VMRD robot grasping dataset [19] with reasonable computation speed of 6.5 frame per second (FPS). However, this modular structure could be further optimized and improved for higher accuracy, faster computation speed and less DNNs for potentially reduced GPU memory usage.

In this paper, we propose a single multi-task DNN with a simple post-processing for OD with reasoning and GD on the piles of novel objects using the information from a single RGB-D camera. Our method is based on YOLOv3 [20] and deals with multi-tasks of OD, GD and relationship reasoning, but maintained its simple single network structure. Ablation studies were performed to further optimize different components of our multi-task networks. Our method yields the state-of-the-art multi-task GD performance (74.2%, 98.6%) on the VMRD and Cornell datasets, respectively, with real-time computation speed (30 and 62 FPS) for high-resolution images of  $608 \times 608$  and  $320 \times 320$  as illustrated in Fig. 1.2. We verify our method to real robotic grasping tasks with a 4 axis robot arm on single novel objects as well as a Baxter robot on multiple novel objects in various settings of piles (cluttered, stacking, invisible scenes) and yielded 95.3% grasp success rate for single novel object grasping and 86.7% grasp success rate in cluttered novel objects, respectively.

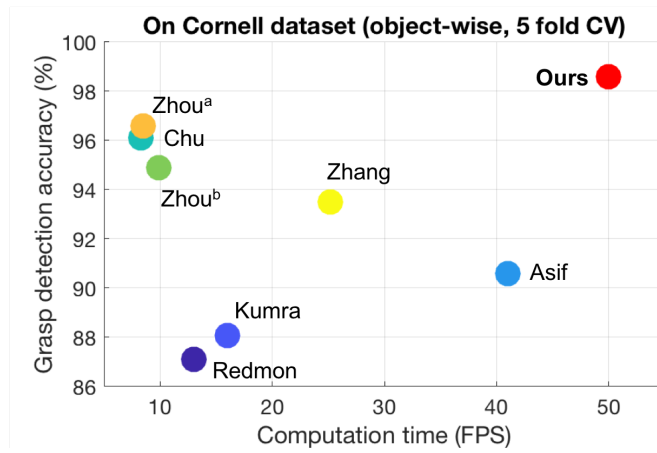


Figure 1.1: REM module performance summary of computation time (frame per second) vs. grasp detection accuracy on the Cornell dataset with object-wise data split.

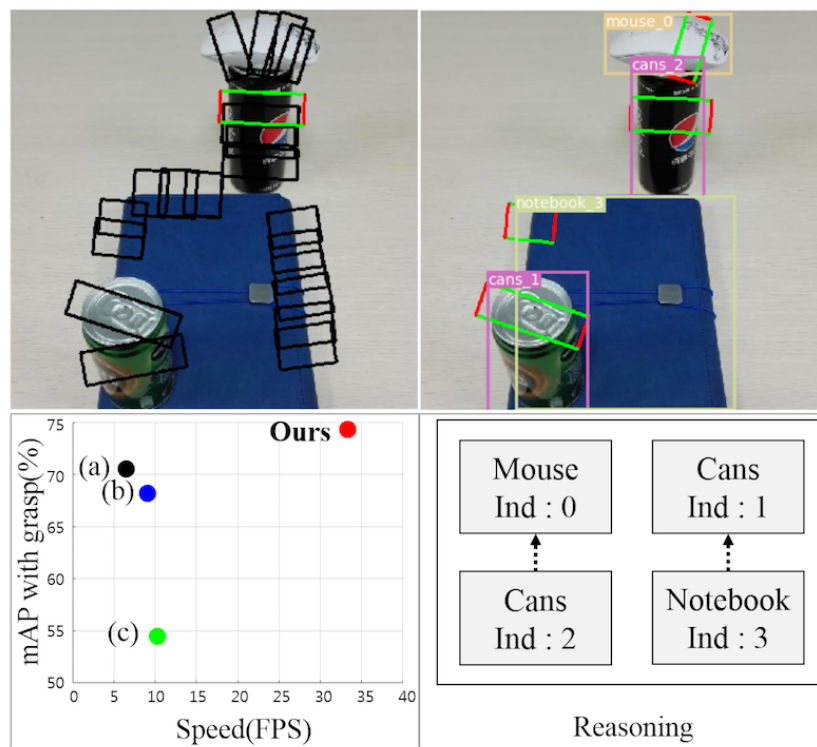


Figure 1.2: (top left panel) GD with grasp candidates (black rectangles) and the best grasp (green and red rectangle) (right panels) multi-tasks of GD, OD and relationship reasoning. (bottom left panel) computation speed (FPS) vs. prediction accuracy (mAP) for multi-task grasping detection of our method achieving the state-of-the-art performance in both accuracy and speed and other previous work of (a) [11], (b,c) [12].

---

# Real-time Highly Accurate Grasping Detection with Rotation Ensemble Module

---

Ideally, robot grasp detection should be rotation-invariant. Rotation angle prediction in robot grasp detection has been done by regression of continuous angle value [13], classification of discretized angles (e.g.,  $10^\circ, 20^\circ, \dots, 170^\circ$ ) [14, 18] or rotation anchor box that is a hybrid method of regression and classification [11, 12, 15]. Previous works were not considering rotation-invariance or attempting rotation-invariant detection by rotating images or feature maps that were often time-consuming especially for multiple objects.

## 2.1 Related works

**Spatial, rotational invariance.** Max pooling layers often alleviate the issue of spatial variance in CNN. To better achieve spatial-invariant image classification, Jaderberg *et al.* proposed spatial transformer network (STN), a method of image (or feature) transformation by learning (affine) transformation parameters so that it can help to improve the performance of inference operations of the following neural network layers [4]. Lin *et al.* proposed to use STN repeatedly with an inverse composite method

by propagating warp parameters rather than images (or features) for improved performance [21]. Esteves *et al.* proposed a rotation-invariant network by replacing the grid generation of STN with a polar transform [10]. Input feature map (or image) was transformed into the polar coordinate with the origin that was determined by the center of mass. Cohen and Welling proposed a method to use group equivariant convolutions and pooling with weight flips and four rotations with  $90^\circ$  stepsize [5]. Follmann *et al.* proposed to use rotation-invariant features that were created using rotational convolutions and pooling layers [6]. Marcos *et al.* proposed a network with a different set of weights for each local window instead of weight rotation [22].

**Object detection.** Faster R-CNN was a method of using a region proposal network for generating region proposals to reduce computation time [23]. YOLO was faster but less accurate than the faster R-CNN by directly predicting  $\{x, y, w, h, \text{class}\}$  without using the region proposal network [24]. YOLO9000 stabilized the loss of YOLO by using anchor box inspired by region proposal network and yielded much faster object detection results than faster R-CNN while its accuracy was comparable [25]. For rotation-invariant object detection, Shi *et al.* investigated face detection using a progressive calibration network that predicted rotation by  $180^\circ$ ,  $90^\circ$  or an angle in  $[-45^\circ, 45^\circ]$  after sliding window [26]. Ma *et al.* used a rotation region proposal network to transform regions for classification using rotation region-of-interest (ROI) pooling [9]. Note that rotation angle was predicted using 1) rotation anchor box, 2) regression or 3) classification.

**Robotic grasp detection.** Deep learning based robot grasp detection methods seem to belong one of the two types: two stage detector (TSD) or one stage detector (OSD). TSD consists of a region proposal network and a detector [11, 12, 14, 15, 18]. After extracting feature maps using proposals from the network in the first stage, objects are detected in the second stage. The region proposal network of TSD generally helps to improve accuracy, but is often time-consuming due to feature map extractions. OSD detects an object on each grid instead of generating region proposal to reduce computation time with decreased prediction accuracy [13]. Lenz *et al.* proposed a TSD model that classifies object graspability using a sparse auto-encode (SAE) with sliding windows for brute-force region proposals [17]. Redmon *et al.* developed a regression based OSD [13] using AlexNet [27]. Guo *et al.* applied ZFNet [28] based TSD to robot grasping and formulated angle prediction as classification [14]. Chu *et al.* further extended the TSD model of Guo [18] by incorporating recent ResNet [29]. Zhou *et al.* also used ResNet for TSD, but proposed rotation anchor box [15]. Zhang *et al.* extended the TSD method of Zhou [15] by additionally predicting objects using ROI [12]. DexNet 2.0 is also TSD that predicts grasp candidates from a depth image and then selects the best one by its classifier, GQ-CNN [30].



## 2.2 Grasping Detection

The goal of the problem is to predict 5D representations for multiple objects from a color image where a 5D representation consists of location  $(x,y)$ , rotation  $\theta$ , width  $w$ , and height  $h$ , as illustrated in Fig. 2.1. Multi-grasp detection often directly estimates 5D representation  $\{x, y, \theta, w, h\}$  as well as its probability (confidence) of being a class (or being graspable)  $z$  for each grid cell. In summary, the 5D representations with its probability are  $\{x, y, \theta, w, h, z\}$ .

For TSD, region proposal networks generate potential candidates for  $\{x, y, w, h\}$  [12, 14, 15, 18] and rotation region proposal network yields possible arbitrary-oriented proposals  $\{x, y, \theta, w, h\}$  [9]. Then, classification is performed for proposals to yield their graspable probabilities  $z$ . Rotation region proposal network classifies rotation anchor boxes with  $30^\circ$  stepsize and then regresses angles.

For OSD, a set of  $\{x, y, \theta, w, h, z\}$  is directly estimated [13]. Inspired by YOLO9000 [25], we propose to use the following reparametrization for 5D grasp representation and its probability for robotic grasp detection as  $\{t^x, t^y, \theta, t^w, t^h, t^z\}$  where  $x = \sigma(t^x) + c_x, y = \sigma(t^y) + c_y, w = p_w \exp(t^w), h = p_h \exp(t^h)$  and  $z = \sigma(t^z)$ . Note that  $\sigma(\cdot)$  is a sigmoid function,  $p_h, p_w$  are the predefined height and width of anchor box, respectively, and  $c_x, c_y$  are the top left corner of each grid cell. Therefore, a DNN directly estimates  $\{t^x, t^y, \theta, t^w, t^h, t^z\}$  instead of  $\{x, y, \theta, w, h, z\}$ .

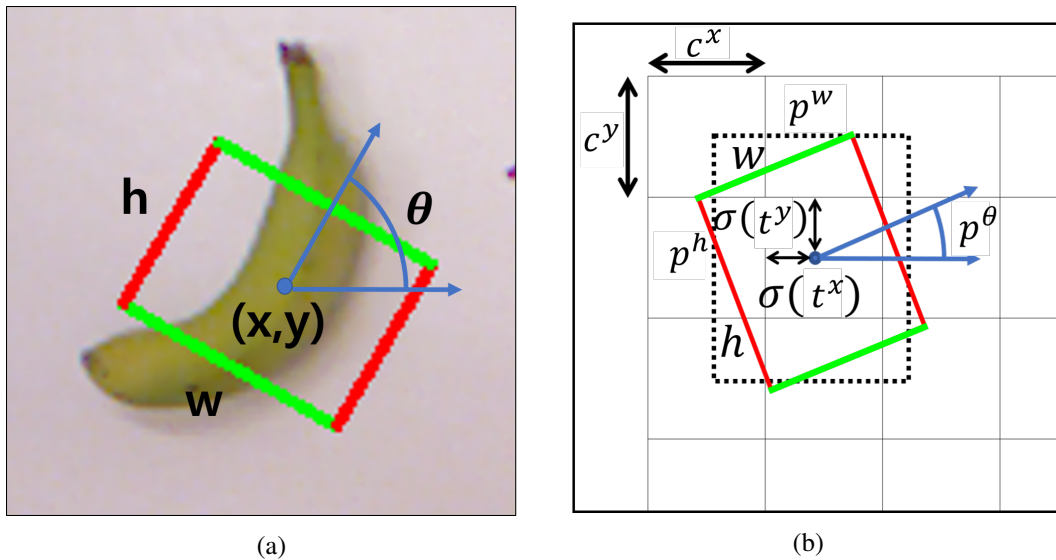


Figure 2.1: (a) A 5D detection representation with location  $(x, y)$ , rotation  $\theta$ , gripper opening with  $w$  and plate size  $h$ . (b) For a  $(2,2)$  grid cell, all parameters for 5D representation are illustrated including a pre-defined anchor box (black dotted box) and a 5D detection representation (red box).

## 2.3 Rotation Ensemble Module

### 2.3.1 Parameter descriptions of the proposed OSD method

For  $S \times S$  grid cells, the following locations are defined

$$(c_x, c_y) \in \{(c_x, c_y) | c_x, c_y \in \{0, 1, \dots, S - 1\}\},$$

which are the top left corner of each grid cell  $(c_x, c_y)$ . Thus, our proposed method estimates the  $(x, y)$  offset from the top left corner of each grid cell. For a given  $(c_x, c_y)$ , the range of  $(x, y)$  will be  $c_x < x < c_x + 1$ ,  $c_y < y < c_y + 1$  due to the reparametrization using sigmoid functions.

We also adopt anchor box approach [25] to robotic grasp detection. Reparametrization changes regression for  $w, h$  into regression & classification. Classification is performed to pick the best representation among all anchor box candidates that were generated using estimated  $t^w, t^h$  and the following  $p_w, p_h$  values:  $\{(0.76, 1.99), (0.76, 3.2), (1.99, 0.76), (1.99, 1.99), (1.99, 3.2), (3.2, 3.2), (3.2, 0.76)\}$  or  $\{(1.99, 1.99)\}$ .

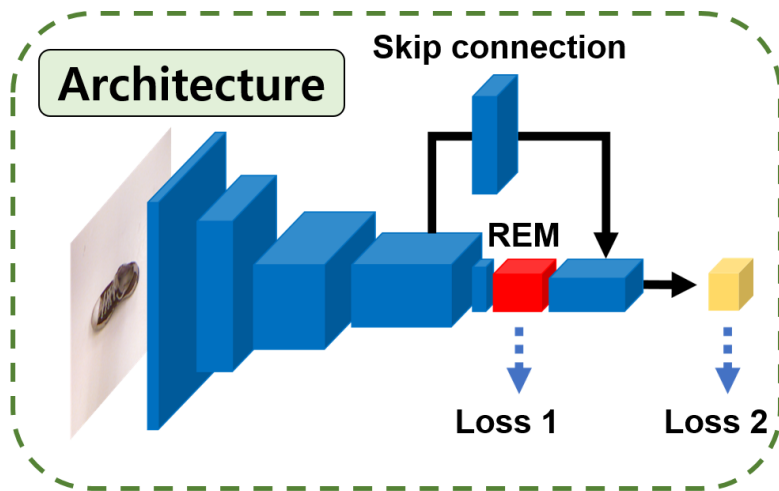
We investigated three prediction methods for rotation  $\theta$ . Firstly, a regressor predicts  $\theta \in [0^\circ, 180^\circ)$ . Secondly, a classifier predicts  $\theta \in \{0^\circ, 10^\circ, \dots, 170^\circ\}$ . Lastly, anchor box approach with regressor & classifier predicts both  $\theta_a \in \{30^\circ, 90^\circ, 150^\circ\}$  and  $\theta_r \in [-30^\circ, 30^\circ]$  to yield  $\theta = \theta_a + \theta_r$ .

Predicting detection (grasp) probability is crucial for multibox approaches such as MultiGrasp [13]. Conventional ground truth for detection probability was 1 (graspable) or 0 (not graspable) [13]. Inspired by [25], we proposed to use IOU (Intersection Over Union) as the ground truth detection probability as  $z^g = |P \cap G| / |P \cup G|$  where  $P$  is the predicted detection rectangle,  $G$  is the ground truth detection rectangle, and  $|\cdot|$  is the area of the rectangle.

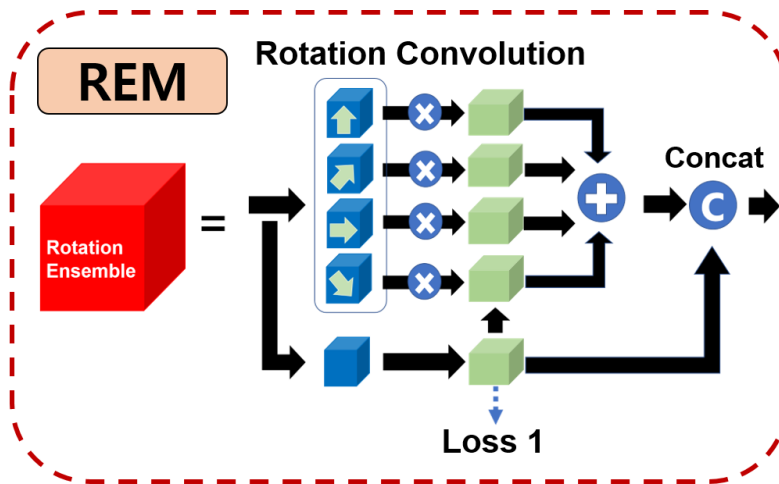
### 2.3.2 Rotation ensemble module (REM)

We propose a rotation ensemble module (REM) with rotation convolution and rotation activation to determine an ensemble weight associated with angle class probability for each grid. We added our REM to the latter part of a robot grasp detection network since it is often effective to put geometric transform related layers in the latter of the network such as deformable convolutions [31]. A typical location for REM in DNNs is illustrated in Fig. 2.2 (a).

Consider a typical scenario of convolution with input feature maps  $f \in \mathbb{R}^{H \times W \times C}$  where  $N = H \times W$  is the number of pixels and  $C$  is the number of channels. Let us denote  $g_l \in \mathbb{R}^{K \times K \times C}$ ,  $l = 1, \dots, n_f$  a convolution kernel where  $K \times K$  is the spatial dimension of the kernel and there are  $n_f$  number of kernels in each channel. Similar to the group convolutions [5], we propose  $n_r$  rotations of the weights to obtain  $n_f \cdot n_r$  rotated weights for each channel. Bilinear interpolations of four adjacent pixel values



(a)



(b)

Figure 2.2: An illustration of incorporating our proposed REM in a DNN for robot grasp detection (a) and the architecture of our proposed REM with rotation convolutions (b).

were used for generating rotated kernels. A rotation matrix is

$$R(r) = \begin{bmatrix} \cos(r\pi/4) & -\sin(r\pi/4) & 0 \\ \sin(r\pi/4) & \cos(r\pi/4) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

where  $r$  is an index for rotations. Then, the rotated weights (or kernels) are  $g_l^i = R(i)g_l, i = 0, \dots, 3, l = 1, \dots, n_f$ . Finally, the output of these convolutional layers with rotation operators for the input  $f$  is

$$d_l^i = g_l^i \star f, i = 0, \dots, 3, l = 1, \dots, n_f,$$

where  $\star$  is a convolution operator. This pipeline of operations is called “rotation convolution”. A typical kernel size is  $K=5$ .

Our REM contains rotation activation that aggregates all feature maps at different angles. Assume that an intermediate output for  $\{t^x, t^y, \theta, t^w, t^h, t^z\}$  is available in REM, called  $\{t_m^x, t_m^y, \theta_m, t_m^w, t_m^h, t_m^z\}$ . Note that  $\theta_m^i \in \mathbb{R}^{H \times W}$  where  $i = 0, \pi/4, 2\pi/4, 3\pi/4$ . For each angle, activations will be generated and all of them must be aggregated to yield one final feature map  $\hat{d}_l = \sum_{i=1}^4 d_l^i \odot \theta_m^i / 4$ . where  $\odot$  is Hadamard product. Thus, our proposed method utilizes class probability (probability to grasp) to selectively aggregate activations along with the weight of angle classification.

In the REM, the intermediate output is partially used for rotation activation, it still contains valuable, compressed information about the final output - it could be a good initial bounding box. Thus, we designed our REM to decompress, concatenate it at the end of REM as illustrated in Fig. 2.2 (b). This pipeline delivers valuable information about  $\{t_m^x, t_m^y, \theta_m, t_m^w, t_m^h, t_m^z\}$  indirectly to the final layer and this structure seemed to decrease probability errors.

## 2.4 Loss functions

We re-designed the loss function for training robotic grasp detection DNNs to emphasize this additional REM. The output of DNN  $(t^x, t^y, \theta, t^w, t^h, t^z)$  and the intermediate output of the REM  $\{t_m^x, t_m^y, \theta_m, t_m^w, t_m^h, t_m^z\}$  should be converted into  $(x, y, \theta, w, h, z)$  and  $\{x_m, y_m, \theta_m, w_m, h_m, z_m\}$ , respectively. Then, using the

ground truth  $(x^g, y^g, \theta^g, w^g, h^g, z^g)$ , the loss function is defined as

$$\begin{aligned} & \lambda_{\text{cd}} (\|m \odot (x - x^g)\|_2 + \|m \odot (y - y^g)\|_2) + \\ & \lambda_{\text{cd}} (\|m \odot (w - w^g)\|_2 + \|m \odot (h - h^g)\|_2) + \\ & \lambda_{\text{pr}} \|m \odot (z - z^g)\|_2 + \lambda_{\text{ag}} \text{AngLoss}(\theta^g, \theta; m) + \\ & \frac{\lambda_{\text{cd}}}{2} (\|m \odot (x_m - x^g)\|_2 + \|m \odot (y_m - y^g)\|_2) + \\ & \frac{\lambda_{\text{cd}}}{2} (\|m \odot (w_m - w^g)\|_2 + \|m \odot (h_m - h^g)\|_2) + \\ & \frac{\lambda_{\text{pr}}}{2} \|m \odot (z_m - z^g)\|_2 + \frac{\lambda_{\text{ag}}}{2} \text{CE}(m \odot \theta^g, m \odot \theta_m) \end{aligned}$$

where  $m$  is a mask vector with 1 (ground truth for that grid) or 0 (no ground truth for that grid),  $\|\cdot\|_2$  is  $l_2$  norm, CE is cross entropy, and AngLoss is one of these functions: CE for classification on  $\theta$ ,  $l_2$  norm for regression or rotation anchor box on  $\theta$ . We chose  $\lambda_{\text{cd}} = \lambda_{\text{ag}} = 1$  and  $\lambda_{\text{pr}} = 5$ .

## 2.5 Simulations and Experiments

We evaluated our proposed REM methods on the Cornell robotic grasp dataset [16, 17] and on real robot grasping tasks with novel objects. The effectiveness of our REM was demonstrated in prediction accuracy, computation time and grasping success rate. Our proposed methods were compared with previous methods such as [12–15, 17, 18] based on literature for widely used Cornell dataset as well as our in-house implementations of some previous works.

### 2.5.1 Implementation details

It is challenging to fairly compare a robot grasp detection method with other previous works such as [12–15, 17, 18]. Due to the Cornell dataset, most works were able to compare their results with those of previous methods that were reported in literature. Considering fast advances of computing power and DNN techniques, it is often not clear how much the proposed scheme or method actually contributed to the increase of performance.

In this paper, we did not only compare our REM methods with previous works on the Cornell dataset through literature, but also implemented the core angle prediction schemes of other previous works with modern DNNs: regression (Reg) that Redmon *et al.* proposed [13], classification (Cls) that Guo *et al.* proposed [14] and rotation anchor box (Rot) that Zhou *et al.* proposed [15]. While Redmon [13], Guo [14] and Zhou [15] used AlexNet [27], ZFNet [28] and ResNet [29], respectively, our in-house implementations, Reg, Cls and Rot, all used DarkNet-19 [32]. While Guo and Zhou were based on faster R-CNN (TSD) [23], our implementations were based on YOLO9000 (OSD) [25].

We performed ablation studies for our REM so that it becomes clear which part will affect the performance of rotated grasp detection most significantly. We placed our proposed REM at the 6th layers from the end of the detection network. We also performed simulations with rotation activation using angle and probability. For multiple robotic grasps detection, boxes were plotted when probabilities were 0.25 or higher.

All algorithms were tested on the platform with GPU (NVIDIA 1080Ti), CPU (Intel i7-7700K 4.20GHz) and 32GB memory. Our REM methods and other in-house DNNs such as Ref, Cls and Rot were implemented with PyTorch.

### 2.5.2 Benchmark dataset and novel objects

The Cornell robot grasp detection dataset [16, 17] consists of 885 images (RGB color and depth) of 240 different objects as shown in Fig. 2.3 with ground truth labels of a few graspable rectangles and a few non-graspable rectangles. We used RG-D information without B channel just like the work of Redmon [13]. An image was cropped to yield a  $360 \times 360$  image and five-fold cross validation was performed. Then, mean prediction accuracy was reported for image-wise and object-wise splits. Image-wise split divides the Cornell dataset into training and testing data with 4:1 ratio randomly without considering the same or different objects. Object-wise is a way of splitting training and testing data with 4: 1 ratio such that both data do not contain the same object. We followed other previous works for accuracy metrics [13, 17, 33]. Successful grasp detection is defined as follows: if IOU is larger than a certain threshold (*e.g.*, 0.25, 0.3 or 0.35) and the difference between the output orientation  $\theta$  and the ground truth orientation  $\theta^g$  is less than  $30^\circ$  (Jaccard index), then it is considered as a successful grasp detection.



Figure 2.3: Images from the Cornell dataset

### 2.5.3 Results for in-house implementations of previous works

Table 2.1 shows the results of ablation studies for our in-house implementations on the Cornell dataset for anchor box with  $w$  and  $h$  with various ratios (N) vs. one ratio of 1:1 (1) and angle prediction methods: regression (Reg) vs. classification (Cls) vs. rotation anchor box (Rot). The results show that using a 1:1 ratio (1) yields better accuracy than using a variety of anchor boxes (N). For angle prediction methods, rotation anchor box yielded the best performance while regression yielded the lowest that was consistent with the literature. Thus, our in-house implementations seem to yield better performance in accuracy than the original previous works possibly due to modern DNNs in our implementations: Reg - Redmon *et al.* [13], Cls - Guo *et al.* [14] and Rot - Zhou *et al.* [15].

Fig. 2.4 shows the results of different angle prediction methods at IOU 25% over epoch. We observed Table 2.1: Ablation studies on the Cornell dataset for anchor box of  $w$ ,  $h$  with various ratios or one ratio and angle prediction methods with Reg, Cls, Rot.

Anchor Box	Angle Prediction	Image-wise		Object-wise	
		25%	35%	25%	35%
N	Reg	91.0	86.5	88.7	85.6
1	Reg	91.8	87.7	89.2	86.3
N	Cls	97.2	93.1	96.1	93.1
<b>1</b>	<b>Cls</b>	<b>97.3</b>	<b>94.1</b>	<b>96.6</b>	<b>92.9</b>
<b>1</b>	<b>Rot</b>	<b>98.3</b>	<b>94.4</b>	<b>96.6</b>	<b>93.6</b>

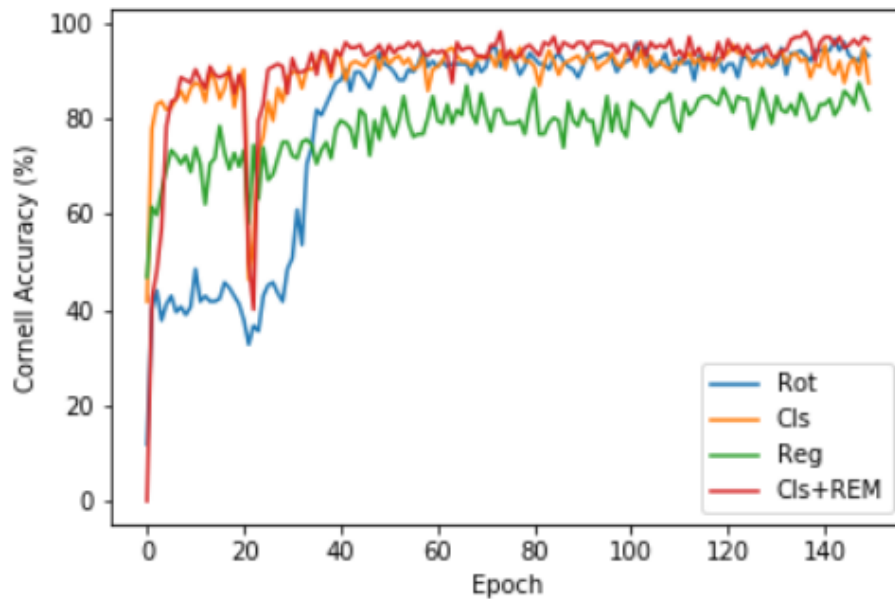


Figure 2.4: Grasp detection accuracy over epoch on the Cornell dataset using various methods for angle predictions: Rot: rotation anchor box, Cls: classification, Reg: regression, REM: ours.

Table 2.2: The ablation studies on the Cornell dataset for our REM with RC, RA and RL.

Angle	RC	RA	RL	Image-wise		Object-wise	
				25%	35%	25%	35%
Cls	-	-	-	97.3	94.1	96.6	92.9
Cls	O	-	-	97.6	94.1	97.3	92.7
<b>Cls</b>	<b>O</b>	<b>O</b>	-	<b>99.2</b>	<b>95.3</b>	<b>98.6</b>	<b>95.5</b>
Cls	O	O	O	98.6	94.9	97.3	94.1
Reg	O	O	-	89.3	84.0	88.3	84.5
Rot	O	O	-	98.5	95.6	98.0	94.0

that Rot yielded slowly increased accuracy over epochs than Cls initially and Reg yielded overall slow increase in accuracy over epochs. These slow initial convergences of Reg and Rot may not be desirable for re-training on additional data.

#### 2.5.4 Results for our proposed REM on the Cornell dataset

Table 2.2 shows the results of the ablation studies for our proposed REM with different components such as rotation convolution (RC) and rotation activation (RA). RA can be obtained by using rotation activation loss (RL) as show in Fig. 2.2. We observed that RC itself did not improve the performance while RC & RA significantly improved the accuracy. Comparable performance was observed when using RC & RA with Rot, but substantially low performance was achieved with Reg.

Table 2.3: Performance summary on Cornell dataset. Our proposed method yielded state-of-the-art prediction accuracy in both image-wise (Img) and object-wise (Obj) splits with real-time computation. The unit for performance is %.

Method	Angle	Type	Img	Obj	Speed (FPS)
			25%	25%	
Lenz [17], SAE	Cls	TSD	73.9	75.6	0.08
Redmon [13], AlexNet	Reg	OSD	88.0	87.1	13.2
Kumra [33], ResNet-50	Reg	TSD	89.2	88.9	16
Asif [34]	Reg	OSD	90.2	90.6	41
Guo [14]#a, ZFNet	Cls	TSD	93.2	82.8	-
Guo [14]#c, ZFNet	Cls	TSD	86.4	89.1	-
Chu [18], ResNet-50	Cls	TSD	96.0	96.1	8.3
Zhou [15]#b, ResNet-50	Rot	TSD	97.7	94.9	9.9
Zhou [15]#a, ResNet-101	Rot	TSD	97.7	96.6	8.5
Zhang [12], ResNet-101	Rot	TSD	93.6	93.5	25.2
<b>Our REM, DarkNet-19</b>	<b>Cls</b>	<b>OSD</b>	<b>99.2</b>	<b>98.6</b>	<b>50</b>

Table 2.3 summarizes all evaluation results on the Cornell robotic grasp dataset for previous works and our proposed methods. Our proposed method yielded state-of-the-art performance, up to 99.2%



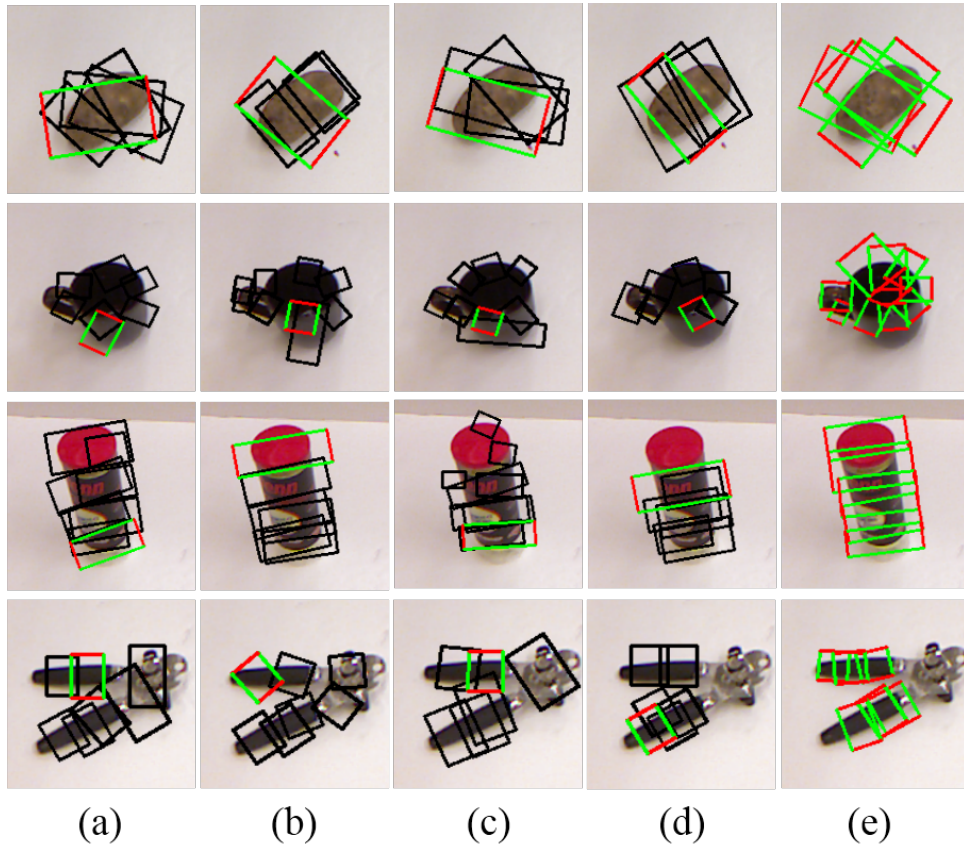


Figure 2.5: Grasp detection results on the Cornell dataset for (a) Reg, a modern version of Redmon [13], (b) Cls, a modern version of Guo [14], (c) Rot, a modern version of Zhou [15] and (d) our proposed Cls+REM. (e) Ground truth labels in Cornell dataset. Black boxes are grasp candidates and green-red boxes are the best grasp among them.

prediction accuracy for image-wise split and up to 98.6% for object-wise split, respectively, over reported accuracies of the previous works that are listed in the Table. Our proposed methods yielded these state-of-the-art performances with real-time computation at 50 frame per second (FPS). Note that AlexNet, DarkNet-19, ResNet-50, ResNet-101 require 61.1, 20.8, 25.6 and 44.5 MB parameters, respectively. Thus, our REM method achieved state-of-the-art results with relatively small size of DNN (20.8MB) compared to other recent works using large DNNs such as ResNet-101 (44.5MB).

Fig. 2.5 illustrates grasp detection results on the Cornell dataset. Our proposed Cls+REM yielded grasp candidates that were close to the ground truth compared to other previous methods such as Reg and Cls.

---

# A Single Neural Network for Multi-Task

---

## 3.1 Related Works

**Pre-deep learning era.** Data-driven GD for novel objects has been investigated extensively [35]. Saxena *et al.* proposed a machine learning (ML) based method to rank the best graspable location for all candidate image patches from different locations [36]. Jiang *et al.* proposed a 5D robotic grasp representation by using a ML method to rank the best graspable image patch whose representation includes orientation and gripper distance among all candidates [37].

**Depth vs color information.** There are several works that use depth information only or color information only for GD. Johns *et al.* developed a method to estimate a grasp score (quality) from a single depth image [38]. Dex-Net 3.0 was proposed to estimate robotic grasps for suction cups from a depth image (point cloud) trained with synthetic data [39]. There have been a couple of works to use depth images only for closed-loop grasping [40, 41]. Morrison *et al.* demonstrated that using fast, lightweight neural network was important for grasping dynamic objects [41]. There also have been some works using color images only for GD. Since depth image is often quite noisy [42], only RGB images have been used for learning 5D grasp representation from a color image [43] and for achieving almost state-of-the-art performance [15].

**OD with GD.** Zhang *et al.* proposed a VMRD grasping dataset with object detection and object relation and a Visual manipulation relationship network (VMRN) [19]. Based on SSD [44], an OD method, VMRN extracted features and then predicted relationship of objects. Zhang *et al.* further developed

multi-task robotic grasp networks for OD, GD and reasoning with VMRN [11, 12] based on the GD work of Zhou [15] for grasping tasks in complex piles of objects.

## 3.2 Multi-task robot grasping

Prior to the challenging grasping operation that requires high-level inference, we first classify it as a sub-problem to solve this problem, and then perform sub-optimization for each related problem first. For this, the subproblems were first defined.

### 3.2.1 Single object robot grasping

A 5D robotic grasp representation is widely used for GD with a parallel gripper when a single 2D image (RGB or RGB-D) is used [17, 37]. This representation is a vector of  $\{x_{gd}, y_{gd}, \theta_{gd}, w_{gd}, h_{gd}\}$  that consists of location  $(x_{gd}, y_{gd})$ , orientation  $\theta_{gd}$ , gripper opening width  $w_{gd}$  and parallel gripper plate size  $h_{gd}$ .

### 3.2.2 Multi-task robot grasping

Grasping a specific target object in cluttered and stacking objects requires more than single object grasping information and needs additional information such as object class and relationship reasoning (see Fig. 1.2) for sequential grasp planning. We extended the 5D robotic grasp representation further to include object class ( $cls_{gd}$ ) and stacking order ( $ord_{gd}$ ) among objects as follows:

$$\{x_{gd}, y_{gd}, \theta_{gd}, w_{gd}, h_{gd}, cls_{gd}, ord_{gd}\}.$$

## 3.3 Reparametrization of 15D representation

We propose a 15D representation for multi-task robot grasping problem to exploit a single multi-task DNN for OD, GD and reasoning altogether. Parameters related to OD are  $\{x_{od}, y_{od}, w_{od}, h_{od}, cls_{od}, pr_{od}\}$  and the parameters related to GD are  $\{x_{gd}, y_{gd}, w_{gd}, h_{gd}, cls_{gd}, pr_{gd}, \theta_{gd}\}$ . where  $pr_{od}$  is a probability of an object existing and  $pr_{gd}$  is a graspable probability. The parameters of reasoning are  $\{cls_{fc}, cls_{cc}\}$  for ordering objects ( $ord_{gd}$ ). Father class (FC) and children class (CC) are labels under and over the predicted target object, respectively. FC and CC are predicted of each grid.

We propose the following reparametrization of OD and GD for robotic grasping in the piles of objects:

$$OD = \{t_{od}^x, t_{od}^y, t_{od}^w, t_{od}^h, t_{od}^{pr}, t_{od}^{cls}\}, R = \{t_{fa}^{cls}, t_{cc}^{cls}\}$$

$$GD = \{t_{gd}^x, t_{gd}^y, t_{gd}^w, t_{gd}^h, t_{gd}^{pr}, t_{gd}^{cls}, t_{gd}^\theta\}$$

where  $x_j = \sigma(t_j^x) + c_j^x, y_j = \sigma(t_j^y) + c_j^y, \sigma(\cdot)$  is a sigmoid function,  $w_j = p_j^w \exp(t_j^w), h_j = p_j^h \exp(t_j^h), \theta_{gd} = p_{gd}^\theta + t_{gd}^\theta, cls_j = \text{softmax}(t_j^{cls}), cls_{rs} = \sigma(t_{rs}^{cls}), pr_j = \sigma(t_j^{pr}), j \in \{od, gd\}$  and  $rs \in \{fc, cc\}$ . Note that  $p_j^h, p_j^w$  and  $p_{gd}^\theta$  are the pre-defined height, width, orientation of an anchor box, respectively, and  $(c_j^x, c_j^y)$  are the location of the top left corner of each grid cell (known). Thus, DNN for GD of our proposed methods will estimate  $\{t_j^x, t_j^y, t_j^\theta, t_j^w, t_j^h, t_j^{pr}\}$  instead of  $\{x_j, y_j, \theta_{gd}, w_j, h_j, pr_j\}$ .  $x_j, y_j, w_j, h_j$  are properly normalized so that the size of each grid is  $1 \times 1$ . Lastly, the angle  $\theta_{gd}$  will be modeled as a discrete and continuous value instead of a continuous value.

### 3.3.1 Anchor box: w, h in each cell

Anchor box approach has been used for OD [25]. Due to re-parametrization with anchor box, estimating  $w_j, h_j$  is converted into estimating  $t_j^w, t_j^h$ , which are related to the expected values of various sizes of  $w_j, h_j$ . Then, the best grasp representation among all anchor box candidates is selected for the final output. Thus, re-parametrization changes regression problem into regression + classification problem for  $w_j, h_j$ . all the configuration of anchor boxes are selected empirically.

### 3.3.2 Anchor box: orientation in each cell.

While MultiGrasp took regression approach for  $\theta_{gd}$  [13], Guo *et al.* converted regression problem of estimating  $\theta_{gd}$  into the classification for  $\theta_{gd}$  among finite number of angle candidates in  $\{0, \pi/18, \dots, 17\pi/18\}$  [14]. Zhang [45] proposed orientation anchor box so that the angle is determined using classification as well as discrete anchor box rotations. Mean average precision (mAP) increased by 3% when using orientation anchor box (4 angles) over angle classification on the VMRD.

### 3.3.3 Object class: cls in each cell.

When objects are stacked in a complex way, it becomes a difficult task to match OD result (detection bounding box) with GD result without additional information such as object classes. For this task, object class is predicted for each of grasp detection box result so that our proposed model can yield grasping detection boxes, their grasping points and corresponding object classes. A softmax was selected for class activation function through our self-evaluation ablation study that will be reported shortly.

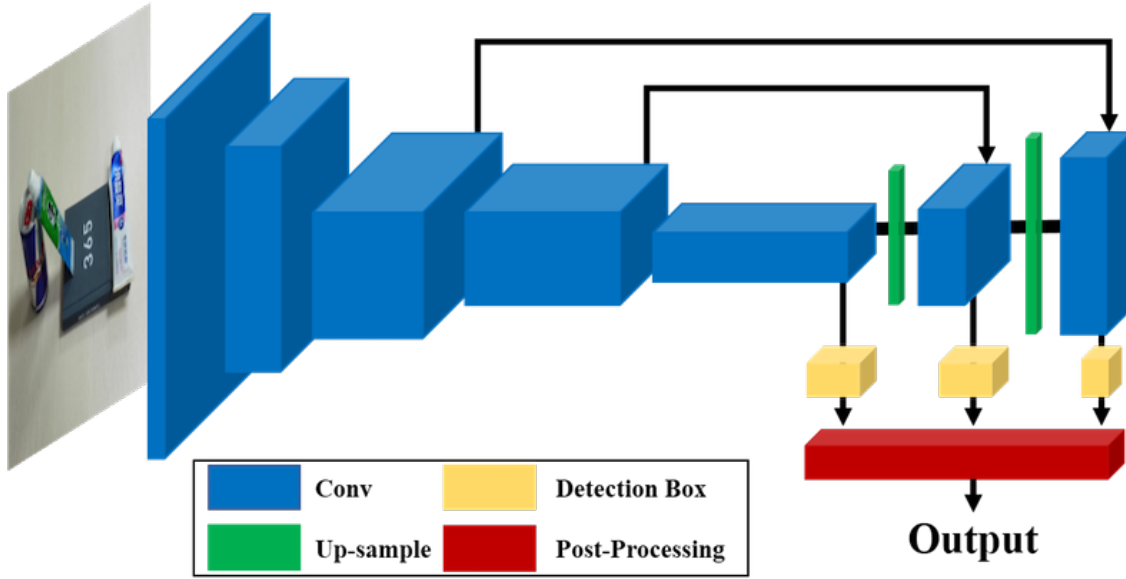


Figure 3.1: Proposed FCNN architecture based on Darknet.

### 3.3.4 FC and CC in each cell.

For inter-object relationship, we propose to predict FC and CC along with other detection results. FC and CC are class labels under and over the target object, respectively. FC and CC consist of object class labels and no-class label.  $(t_{fc}^{cls}, t_{cc}^{cls}) \in \{class_1, class_2, \dots, class_{none}\}$ . In our experiment, we observed that CC is more accurately estimated than FC. Thus, we only used CC for reasoning for the best possible results.

## 3.4 Proposed FCNN with predictions across scales

Our proposed FCNN inherited pre-trained Darknet-53 of YOLOv3 for OD [20] and extended it for multi-task OD with reasoning and GD as illustrated in Fig. 3.1. For our multi-task predictions, we did not only adopted prediction across scales for OD using feature pyramid networks [46], but also extended it for reasoning and GD.

On the low-resolution scale, three anchor boxes  $(w, h)$  for OD 1 anchor box for GD and 4 anchor boxes for grasping angles are predicted as

$$(p_{od}^w, p_{od}^h) \in \{(540, 540), (480, 480), (420, 420)\},$$

$$(p_{gd}^w, p_{gd}^h) \in \{(300, 300)\}, p_{gd}^\theta \in \{0, \pi/4, 2\pi/4, 3\pi/4\}.$$

Then, on the mid-resolution scale after  $\times 2$  bilinear up-sampling, 3 anchor boxes for OD, 1 anchor box

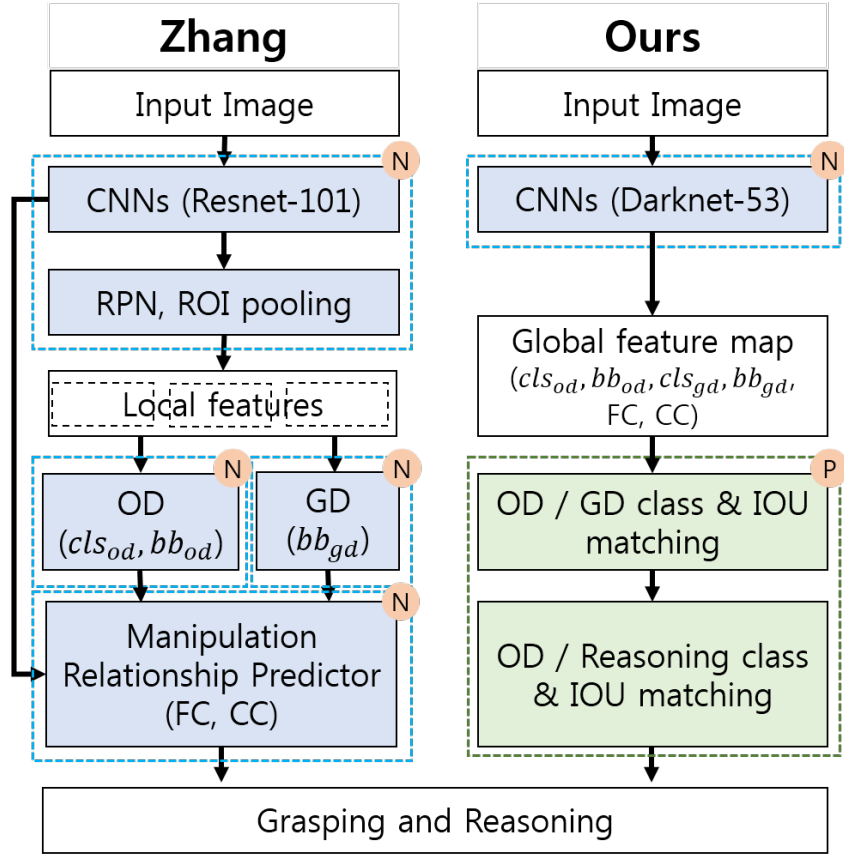


Figure 3.2: Schematic pipelines of Zhang [11, 12] vs ours. Mark 'N' means neural network and 'P' means post-processing.

for GD and 4 anchor boxes for grasping angles are estimated as

$$(p_{od}^w, p_{od}^h) \in \{(360, 360), (300, 300), (240, 240)\},$$

$$(p_{gd}^w, p_{gd}^h) \in \{(100, 100)\}, p_{gd}^\theta \in \{0, \pi/4, 2\pi/4, 3\pi/4\}.$$

On the high-resolution scale after  $\times 4$  up-sampling, the following anchor boxes for our multi-tasks are predicted:

$$(p_{od}^w, p_{od}^h) \in \{(180, 180), (120, 120), (60, 60)\}.$$

OD with reasoning are performed across scales of  $\times 1$ ,  $\times 2$  and  $\times 4$  and GD are performed across scales of  $\times 1$  and  $\times 2$ . Therefore, 9 anchor boxes are predicted with 4 bounding box offsets, object probability, object class (class number) and  $(\text{class number}+1)\times 2$  reasoning classes (FC, CC) for OD with reasoning. In addition, 8 anchor boxes are predicted with 4 bounding box offsets, orientation, grasp probability and object class (class number) for GD.

### 3.5 Reasoning post-processing: from class to index

Fig. 3.2 illustrates the differences between the works of Zhang [11, 12] and our proposed methods. Previous work generated necessary information for OD with reasoning and GD. Deep neural networks (N) generated local features or OD or GD or relationship among objects (FC, CC), respectively and sequentially. However, as shown in [13], dealing with GD and classification often improves the overall performance of GD. We propose a novel single network (N) to yield most information on OD and GD with simple reasoning post-processing (P) for building hierarchy among objects.

For a generated global feature map including class information, bounding box information and FC / CC, reasoning post-processing can build index relationships using class information. Firstly, non-maximum suppression is applied to GD and OD to eliminate unnecessary detection results. Secondly, bounding boxes in OD and GD ( $bb_{od}$ ,  $bb_{gd}$ ) are grouped based on their class information ( $cls_{od}$ ,  $cls_{gd}$ ). Then, the spatial information of bounding boxes are used for further grouping bounding box pairs for OD and GD based on the IOU (Intersection Over Union) as follows:

$$IOU = \frac{bb_{od} \cap bb_{gd}}{bb_{od} \cup bb_{gd}}. \quad (III.1)$$

Lastly, among GD candidates whose IOU exceeds a certain threshold, the best probability for GD is selected to obtain the final OD / GD bounding box pair. Similarly, we compare object classes with child classes already obtained in the model to get the relationships between them by matching boxes with IOU threshold. With these technologies, we could make a object relation graph for robot grasping.

### 3.6 Loss functions

For the output vectors OD, GD and R of DNN and the ground truth (GT)  $OD_{gt}$ ,  $GD_{gt}$  and  $R_{gt}$ , we propose the following loss function to train our single multi-task DNN:

$$\begin{aligned} & \sum_{i \in \Omega} \sum_{j \in \{od, gd\}} \left\{ \sum_{k \in \{x, y, w, h\}} \text{MSE}(k_j^i, k_{j,gt}^i) + \right. \\ & \left. \sum_{k \in pr} (-\log k_j^i) + \sum_{k \in cls_{ob}} \text{FocLoss}(k_j^i, k_{j,gt}^i) \right\} + \\ & \lambda_n \sum_{i \in \Omega^c} \sum_{j \in \{od, gd\}} \sum_{k \in pr} (-\log(1 - k_j^i)) + \\ & \sum_{i \in \Omega} \sum_{j \in R} \sum_{k \in cls_{fc}, cls_{cc}} \text{FocLoss}(k_j^i, k_{j,gt}^i) + \\ & \sum_{i \in \Omega} \sum_{j \in gd} \sum_{k \in \theta} \text{MSE}(k_j^i, k_{j,gt}^i) \end{aligned} \quad (III.2)$$

where  $x, y, w, h, z$  are functions of  $t^x, t^y, t^w, t^h, t^z$  respectively,  $\Omega$  is the grid cells where the object or grasping object are located. Since  $cls_{fc}$  and  $cls_{cc}$  are multi-classes and imbalance, we used focal loss developed for training highly accurate dense object detectors (FocLoss) [47]:

$$FocLoss(p_t) = -(1 - p_t)^\gamma \log p_t \quad (\text{III.3})$$

Focal loss gamma is set to 2 and we set  $\lambda_n = 100$ .

## 3.7 Experimental Evaluations

We evaluated our proposed methods on the VMRD dataset [11], the Cornell dataset [16].

### 3.7.1 Implementation details

Darknet-53 was also implemented for the evaluations on the VMRD and for real multi-task robot grasping of multi objects. Either stochastic gradient descent (SGD) with momentum of 0.9 or Adam optimizer was used for training. Learning rate was 0.001 and mini batch size was set to 2. For self-evaluation to optimize the model, total epoch was 50. Once the model is optimized, total epoch was set to 100 with reducing learning rate by half every 30 epochs. Patch based training was performed with the sizes of  $608 \times 608$  using data augmentation [11]. All algorithms were tested on the platform with a single GPU (NVIDIA GTX1080Ti), a single CPU (Intel i7-7700K 4.20GHz) and 32GB memory.

### 3.7.2 Evaluations on VMRD and Cornell datasets

We performed benchmarks using the Cornell dataset [16, 17] as illustrated in Fig. ???. This dataset consists of 855 images (RGB-D) of 240 different objects with GT labels of a few graspable / non-graspable rectangles. We cropped images with  $360 \times 360$ , but did not resize it to  $224 \times 224$ . Five-fold cross validation (CV) was performed and average prediction accuracy was reported for image-wise and object-wise splits. When the difference between the output orientation  $\theta$  and the GT orientation  $\theta_{gt}$  is less than a certain threshold (e.g.,  $30^\circ$ ), then IOU that is larger than a certain threshold (e.g., 0.25, 0.3) will be considered as a successful grasp detection. The same metric for accuracy has been used in previous works [12–15, 17, 18, 33, 48].

VMRD dataset was used to train our single multi-task network. VMRD consists of 4233 train data and 450 test data (RGB images) as illustrated in Fig. 3.3. In this dataset, there are 2-5 objects stacked in each image and GT for OD with with class label & relationship index, GD with class label and FC / CC labels. There are 31 object classes. If the IOU for predicted OD and GT OD is larger than 0.5 and the



best grasping point for that object meets the above Cornell evaluation metric, it is considered as success (mAP with grasp or mAPg) [11].

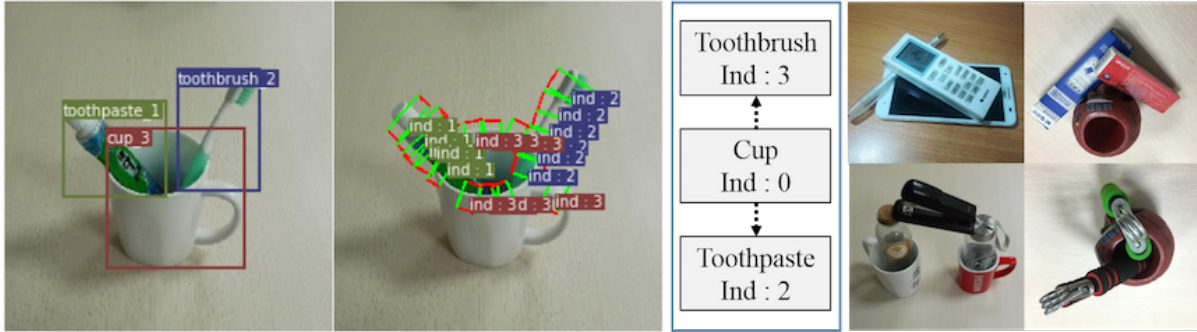


Figure 3.3: VMRD dataset.

## 3.8 Results

### 3.8.1 Simulation results on VMRD dataset

Table 3.1: Self-evaluation summary on VMRD.

Across scales	Activation	Loss	Opt.	mAPg (%)
1, 2, 3	Sigmoid	Cross Entropy	Adam	56.5
1, 2, 3	Softmax	Cross Entropy	Adam	63.1
1, 2	Softmax	Cross Entropy	Adam	64.9
1, 2	Softmax	Focal Loss	Adam	67.1
1, 2	Softmax	Focal Loss	SGD	<b>69.2</b>

Table 3.1 summarizes our ablation study results on the VMRD multi-task robot grasp dataset. The method on the first row of Table 3.1 is an initial extension of YOLOv3 to multi-task robot grasping. Then, by changing activation function, scale, loss function and optimization algorithm, we were able to optimize our single DNN empirically for multi-task OD, GD and reasoning from 56.5% mAPg (mAP with grasp) to 69.2% mAPg. The VMRD dataset seems unbalanced since there are 2061 notebooks and 93 chargers. Focal Loss gives small weights to well-classified examples while gives large weights to some examples that are difficult to classify to focus on learning difficult examples.

Fig. 3.4 illustrates qualitative results for generating multi-task robotic grasps. Fig. 3.4(a) shows a two-level stacking case and its OD, GD and reasoning results of our proposed method (bottom row) and GT (top row). Fig. 3.4(b) shows another multi-stacking case of GT (top) and the output of our proposed method (bottom). Note that GT contains an error in reasoning (Stapler is not on the Apple) while our method corrected for it through training on many examples.

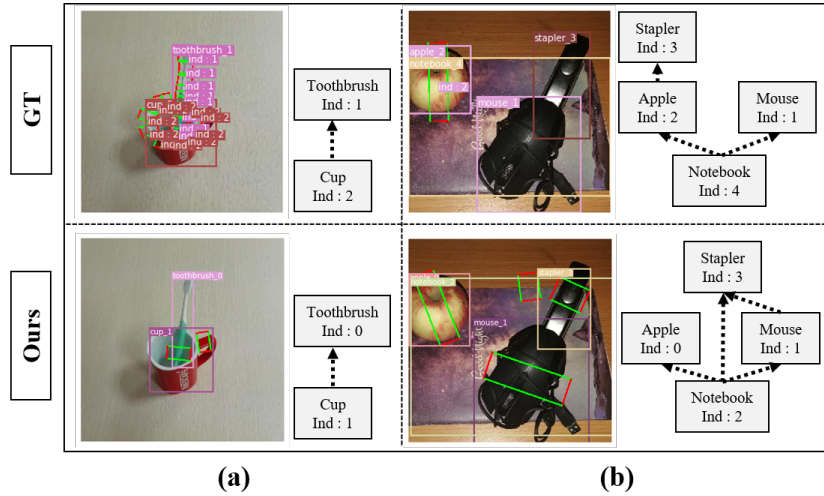


Figure 3.4: Multi-task detection results for VMRD. The 1<sup>st</sup> row is GT and the 2<sup>nd</sup> row is the results of our proposed methods. Note that our method yielded correct reasoning result for “Stapler” while GT incorrectly describes it.

Table 3.2 summarizes the results of the results of previous methods [11, 12] and our proposed method. Our proposed method yielded state-of-the-art performance of 74.3% mAP with grasp (mAPg) at the fastest computation speed of 33.3 FPS for a high resolution input image (608×608).

Table 3.2: Performance summary on VMRD dataset.

Method	mAPg (%)	Speed (FPS)
Zhang [12] baseline, OD, GD	54.5	10.3
Zhang [12], OD, GD	68.2	9.1
Zhang [11], OD, GD, reasoning	70.5	6.5
Ours, OD, GD, reasoning	74.6	33.3

### 3.8.2 Simulation results on Cornell dataset

Fig. 4.1 illustrates qualitative results for generating robotic grasps using our methods without and with predictions across scales. Both yielded fairly good grasp detection results, but there were often cases with fine details where predictions across scales improved the results such as the case with scissors as shown in Fig.4.1.

Table 3.3 summarizes the results of previous methods and our methods. Our proposed method with RGB-D yielded state-of-the-art performance of up to 98.6% prediction accuracy for image-wise split and up to 97.2% for object-wise split, respectively. Our proposed method with RGB also yielded comparable results to state-of-the-art methods. Note that our proposed method yielded these results with

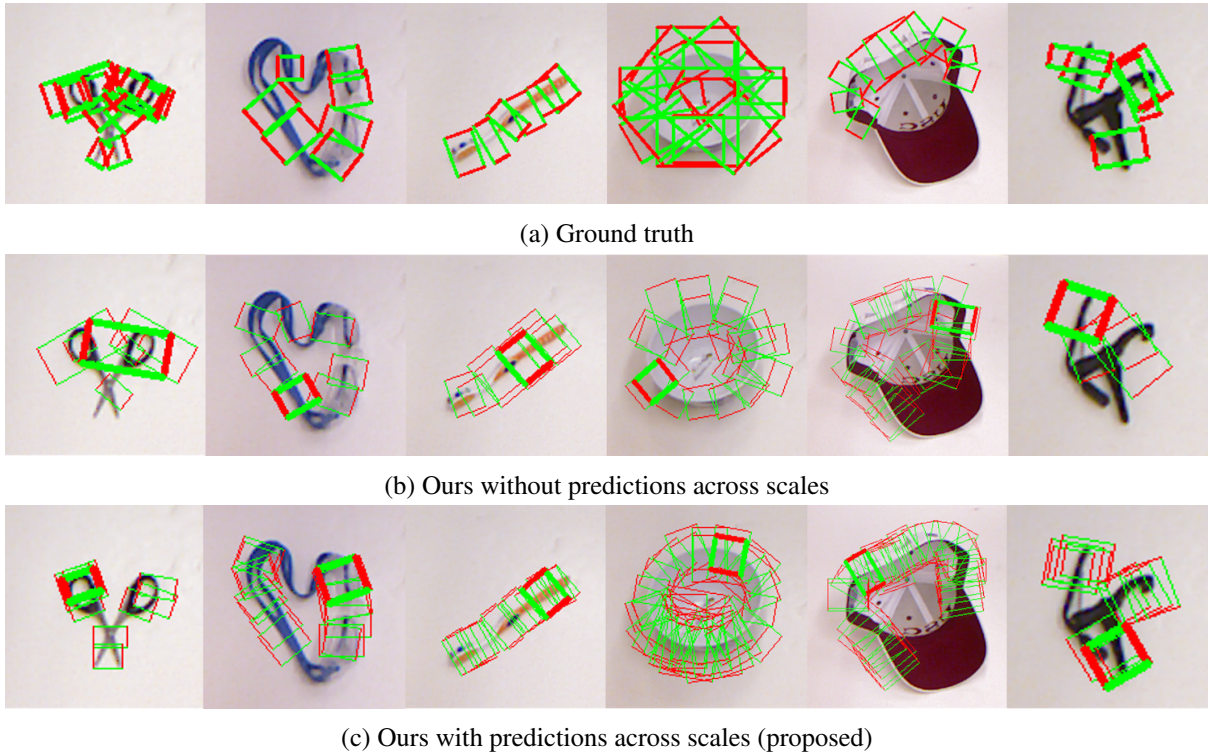


Figure 3.5: GD results on Cornell dataset using our methods without and with predictions across scales.

the smallest DNN and the fastest computation time of 16 ms per high resolution image ( $360 \times 360$ ) that can be potentially useful for real-time applications or stand-alone applications with limited memory and energy. Using depth and predictions across scales improved performance.

Table 3.3: Summary on Cornell data (25% IOU).

Method	Input	Image (%)	Object (%)	Speed (FPS)
Lenz [17], SAE	RGB-D	73.9	75.6	0.08
Redmon [13], Alexnet	RG-D	88.0	87.1	13.2
Kumra [33], Resnet-50	RGB-D	89.2	88.9	16
Asif [48]	RGB-D	90.2	90.6	41
Guo [14] #a, ZFnet	RGB-D	93.2	82.8	-
Guo [14] #c, ZFnet	RGB-D	86.4	89.1	-
Chu [18], Resnet-50	RG-D	96.0	96.1	8.3
Zhou [15], Resnet-50	RGB	97.7	94.9	9.9
Zhou [15], Resnet-101	RGB	97.7	96.6	8.5
Zhang [12], Resnet-101	RGB	93.6	93.5	25.2
Ours, Darknet-19	RGB	97.7	96.1	<b>140</b>
Ours, Darknet-19	RG-D	<b>98.6</b>	<b>97.2</b>	62.5

---

# Toward Robot Demonstration in Real-environment

---

## 4.1 Evaluation of multi-tasks OD, GD, reasoning with Baxter

We evaluated our proposed methods using a Baxter with 7-axis arms (Rethink Robotics, Germany, see Fig. 4.1a) for three different scenarios of cluttered scene, stacking scene and complex invisible stacking scene. In cluttered scene, it was recorded as success if the robot grasped the target in a single try. In stacking and invisible scenes, it was recorded as success if the robot removed objects over the target and then grasp the target. For the invisible scene with no target detected, the robot put away overlapping objects one by one until the target is found. Prediction was performed separately between all robot movements. All combinations of items, the target object and stacking orders are chosen randomly.

## 4.2 Evaluation of GD with 4-axis robot arm

We evaluated our proposed methods with a small 4-axis robot arm (Dobot Magician, Shenzhen YueJiang Tech, China) for novel object grasping. The following 8 novel objects (toothbrush, candy, earphone cap, cable, styrofoam bowl, L-wrench, nipper, pencil) were used for grasping tasks. If the robot gripper grasps an object and moves the object to another place, it is counted as success.

Table 4.1: Performance summary of grasping tasks for cluttered (CS), stacking (SS) and invisible (IS) scenes.

#objects	2	3	4	5
CS	-	86.7(13/15)	85.0%(17/20)	86.7%(26/30)
SS	80.0%(8/10)	60.0%(9/15)	55.0%(11/20)	-
IS	-	60.0%(9/15)	40.0%(8/20)	28.0%(7/25)

### 4.3 Results of Robot Evaluation in Real-environment

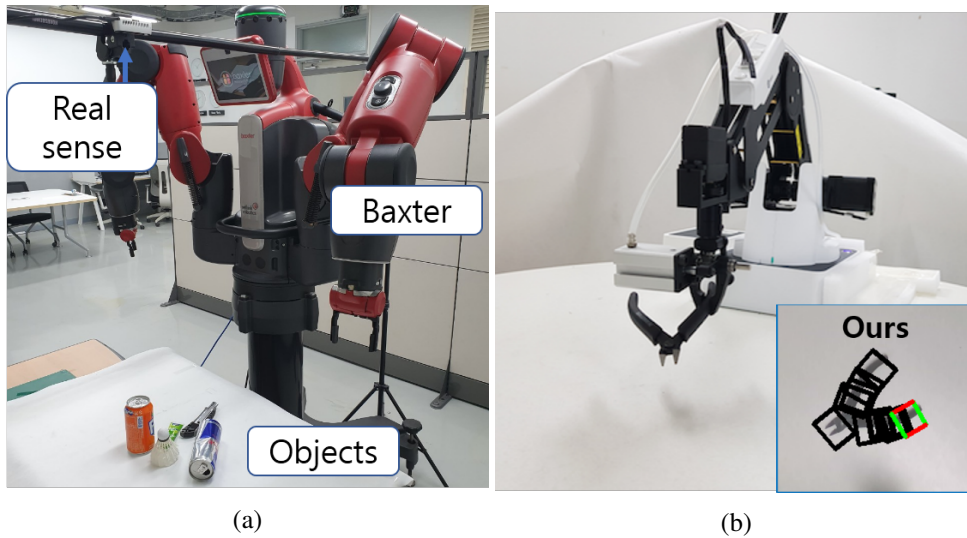


Figure 4.1: (a) our real multi-task evaluation environment (Baxter). (b) our robot grasping experiment with 4-axis robot.

#### 4.3.1 Results of multi-task OD, GD, reasoning with Baxter

Fig. 4.2a shows the OD, GD and reasoning results of our proposed methods for different scenarios such as (a) cluttered scene (CS), (b) stacking scene (SS) and (c) invisible scene (IS). For CS, the target “Stapler” was successfully located with proper grasp. For SS, the target “Knife” and its related object “Toothpaste” were well located with correct relationship reasoning. For IS, the target was not detected due to occlusion, but as overlapped objects are removed based on the reasoning results (green arrow), the target was finally detected at the step 3. Table 4.1 shows the performance summary of the results of our proposed method with a Baxter robot. In CS, the accuracy was high, up to 86.7% regardless of the increase in the number of objects. However, in SS, we observed that increasing the number of objects decreases grasp success rate possibly due to the difficulties of FC, CC predictions among them with severe occlusions. This phenomenon was also observed in challenging IS case. Fig. 4.2b show that the results when the items are stacked. We target the knife at first step. After building the relationship

among those objects, then firstly grasped the toothpaste and then re-try whole detection process and conduct grasping. However in the stacking scene, we found that increasing the number of objects had a significant effect on the accuracy, thus making a lot of changes in the child and father class predictions. Fig. 4.2c show that the results when target item is invisible. It also can be demonstrated well(60%) but the same effect appear.

### 4.3.2 Results of GD with 4-axis robot arm

Fig. 4.1b illustrates our robot grasp experiment with “nipper”. Note that due to small gripper and small objects, grasp detection accuracy was important for successful robot grasping. Our proposed method yielded 95.3% mean grasp success rate with 6.5% standard deviation for 8 novel, small objects with 8 repetitions per each object.

## 4.4 Discussion

It was confirmed that the work was done well as long as the number of objects was limited by putting off the work performed after the simple calibration work. However, in the stacking and invisible scenes, it could be confirmed that the stack relationship of the object was still not fully inferred, and it is believed that a way to improve the post-processing stage for the Real-environment manipulation part could be further researched.

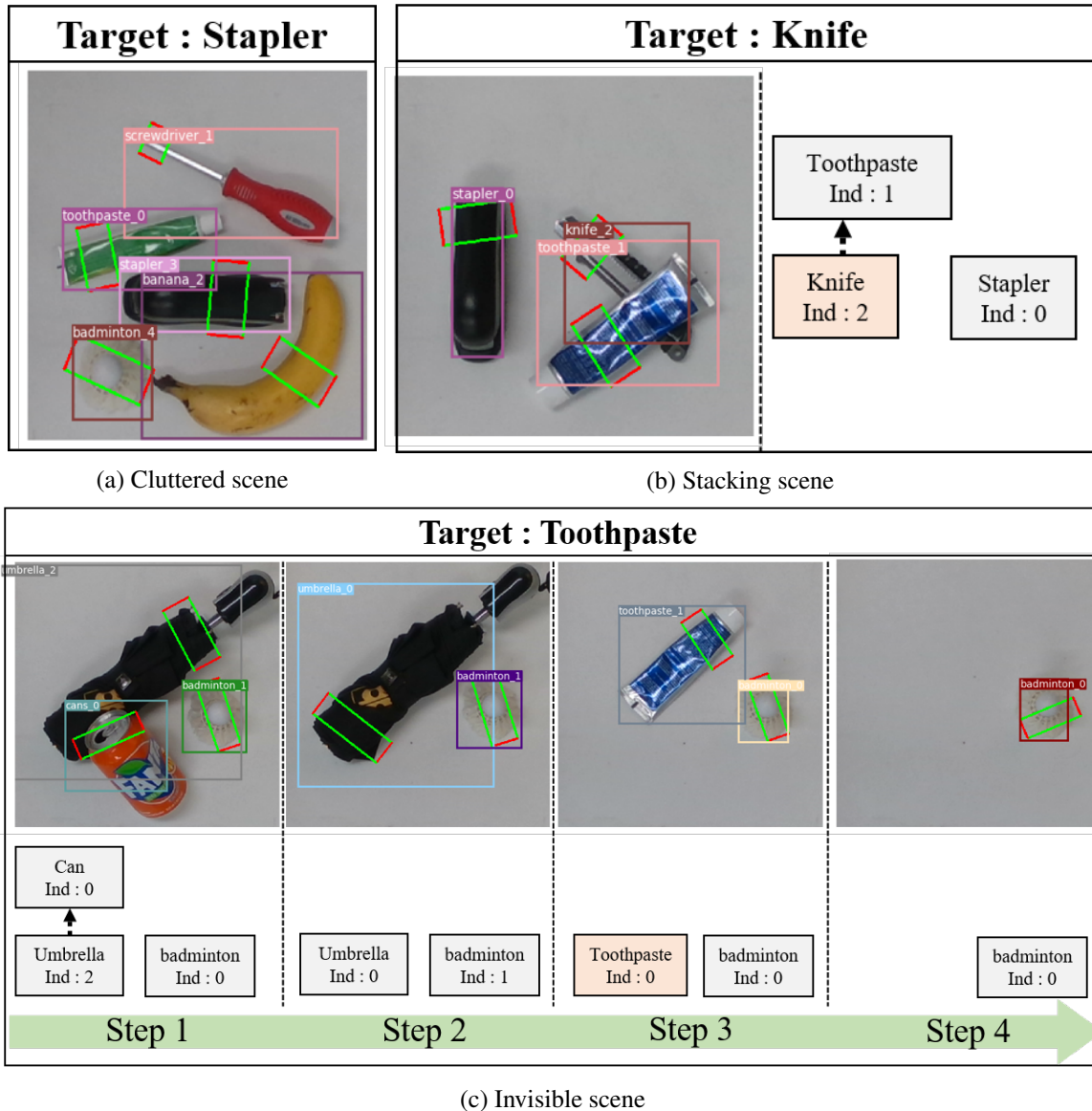


Figure 4.2: Target grasp detection results in (a) cluttered scene, (b) stacking scene and (c) challenging invisible scene.



---

## Conclusion

---

We propose the REM for robotic grasp detection that was able to outperform state-of-the-art methods by achieving up to 99.2% (image-wise), 98.6% (object-wise) accuracies on the Cornell dataset with fast computation (50 FPS) and reliable grasps for multi-objects [49].

We propose a single multi-task DNN that yields the information on GD, OD and reasoning among objects with a simple post-processing [50]hods yielded state-of-the-art performance with the accuracy of 98.6% and 74.2% and the computation speed of 33 and 62 FPS on VMRD and Cornell datasets, respectively. Our methods also yielded 95.3% grasp success rate for single novel object grasping with a 4-axis robot arm and 86.7% grasp success rate in cluttered novel objects with a humanoid robot. However, we still got limitation, the multi-task dataset to which our algorithm fitted has many noisy data and then it could lower the grasping accuracy in real grasping tasks.

But there are many remaining things to do for ultimate grasping. From our experiments, we found that the direction in which the robot is driven and the final real robot processing is also affected by the actual real environment factors such as unseen obstacles and hidden hinders. So, a research for solving it should be focusing on the final operation of the robot which containing the orientation information and actions with joints in real-time and keep interacts with real-environment via many sensors.

---

## References

---

- [1] Henry A Rowley, Shumeet Baluja, and Takeo Kanade, “Rotation invariant neural network-based face detection,” Tech. Rep., CARNEGIE-MELLON UNIV PITTSBURGH PA SCHOOL OF COMPUTER SCIENCE, 1997. [i](#)
- [2] Hayit Greenspan, Serge Belongie, Rodney Goodman, Pietro Perona, Subrata Rakshit, and Charles H Anderson, “Overcomplete steerable pyramid filters and rotation invariance,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1994, pp. 222–228. [i](#)
- [3] Whoi-Yul Kim and Po Yuan, “A practical pattern recognition system for translation, scale and rotation invariance,” in *1994 Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 1994, pp. 391–396. [i](#)
- [4] M Jaderberg, K Simonyan, A Zisserman, and Koray Kavukcuoglu, “Spatial transformer networks,” in *Advances in Neural Information Processing Systems 28*, 2015, pp. 2017–2025. [i](#), [4](#)
- [5] Taco Cohen and Max Welling, “Group equivariant convolutional networks,” in *International Conference on Machine Learning (ICML)*, 2016, pp. 2990–2999. [i](#), [5](#), [7](#)
- [6] Patrick Follmann and Tobias Bottger, “A rotationally-invariant convolution module by feature map back-rotation,” in *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2018, pp. 784–792. [i](#), [5](#)

## REFERENCES

- 
- [7] Fisher Yu and Vladlen Koltun, “Multi-scale context aggregation by dilated convolutions,” in *ICLR*, 2016. [i](#)
- [8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, “Spatial pyramid pooling in deep convolutional networks for visual recognition,” in *European conference on computer vision*, 2014, pp. 346–361. [i](#)
- [9] Jianqi Ma, Weiyuan Shao, Hao Ye, Li Wang, Hong Wang, Yingbin Zheng, and Xiangyang Xue, “Arbitrary-oriented scene text detection via rotation proposals,” *IEEE Transactions on Multimedia*, 2018. [i](#), [5](#), [6](#)
- [10] Carlos Esteves, Christine Allen-Blanchette, Xiaowei Zhou, and Kostas Daniilidis, “Polar transformer networks,” in *International Conference on Learning Representations (ICLR)*, 2018. [i](#), [5](#)
- [11] Hanbo Zhang, Xuguang Lan, Lipeng Wan, Chenjie Yang, Xinwen Zhou, and Nanning Zheng, “Rprg: Toward real-time robotic perception, reasoning and grasping with one multi-task convolutional neural network,” *arXiv preprint arXiv:1809.07081*, 2018. [iv](#), [2](#), [3](#), [4](#), [5](#), [16](#), [19](#), [20](#), [21](#), [22](#), [23](#)
- [12] Hanbo Zhang, Xuguang Lan, Xinwen Zhou, and Nanning Zheng, “Roi-based robotic grasp detection in object overlapping scenes using convolutional neural network,” *arXiv preprint arXiv:1808.10313*, 2018. [iv](#), [3](#), [4](#), [5](#), [6](#), [10](#), [13](#), [16](#), [19](#), [20](#), [21](#), [23](#), [25](#)
- [13] J Redmon and A Angelova, “Real-time grasp detection using convolutional neural networks,” in *IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 1316–1322. [iv](#), [2](#), [4](#), [5](#), [6](#), [7](#), [10](#), [11](#), [12](#), [13](#), [14](#), [17](#), [20](#), [21](#), [25](#)
- [14] Di Guo, Fuchun Sun, Huaping Liu, Tao Kong, Bin Fang, and Ning Xi, “A hybrid deep architecture for robotic grasp detection,” in *IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 1609–1614. [iv](#), [2](#), [4](#), [5](#), [6](#), [10](#), [12](#), [13](#), [14](#), [17](#), [21](#), [25](#)
- [15] Xinwen Zhou, Xuguang Lan, Hanbo Zhang, Zhiqiang Tian, Yang Zhang, and Nanning Zheng, “Fully convolutional grasp detection network with oriented anchor box,” in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 7223–7230. [iv](#), [1](#), [2](#), [4](#), [5](#), [6](#), [10](#), [12](#), [13](#), [14](#), [15](#), [16](#), [21](#), [25](#)
- [16] Ian Lenz, Honglak Lee, and Ashutosh Saxena, “Deep Learning for Detecting Robotic Grasps,” in *Robotics: Science and Systems*, June 2013, p. P12. [1](#), [2](#), [10](#), [11](#), [21](#)

## REFERENCES

- [17] Ian Lenz, Honglak Lee, and Ashutosh Saxena, “Deep learning for detecting robotic grasps,” *The International Journal of Robotics Research*, vol. 34, no. 4-5, pp. 705–724, Apr. 2015. [1](#), [2](#), [5](#), [10](#), [11](#), [13](#), [16](#), [21](#), [25](#)
- [18] Fu-Jen Chu, Ruinian Xu, and Patricio A Vela, “Real-World Multiobject, Multigrasp Detection,” *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3355–3362, Oct. 2018. [2](#), [4](#), [5](#), [6](#), [10](#), [13](#), [21](#), [25](#)
- [19] Hanbo Zhang, Xuguang Lan, Xinwen Zhou, Zhiqiang Tian, Yang Zhang, and Nanning Zheng, “Visual manipulation relationship network for autonomous robotics,” in *2018 IEEE-RAS 18th International Conference on Humanoid Robots (Humanoids)*. IEEE, 2018, pp. 118–125. [2](#), [15](#)
- [20] Joseph Redmon and Ali Farhadi, “Yolov3: An incremental improvement,” *arXiv preprint arXiv:1804.02767*, 2018. [2](#), [18](#)
- [21] Chen-Hsuan Lin and Simon Lucey, “Inverse compositional spatial transformer networks,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. [5](#)
- [22] Diego Marcos, Michele Volpi, Nikos Komodakis, and Devis Tuia, “Rotation equivariant vector field networks,” in *IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 5058–5067. [5](#)
- [23] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun, “Faster R-CNN: Towards real-time object detection with region proposal networks,” in *Advances in neural information processing systems (NIPS)*, 2015, pp. 91–99. [5](#), [10](#)
- [24] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi, “You Only Look Once: Unified, Real-Time Object Detection,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 779–788. [5](#)
- [25] Joseph Redmon and Ali Farhadi, “YOLO9000: Better, Faster, Stronger,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 6517–6525. [5](#), [6](#), [7](#), [10](#), [17](#)
- [26] Xuepeng Shi, Shiguang Shan, Meina Kan, Shuzhe Wu, and Xilin Chen, “Real-time rotation-invariant face detection with progressive calibration networks,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 2295–2303. [5](#)
- [27] A Krizhevsky, I Sutskever, and G E Hinton, “Imagenet classification with deep convolutional neural networks,” in *Advances in Neural Information Processing Systems 25*, 2012, pp. 1097–1105. [5](#), [10](#)

## REFERENCES

- 
- [28] Matthew D Zeiler and Rob Fergus, “Visualizing and understanding convolutional networks,” in *European Conference on Computer Vision (ECCV)*, 2014, pp. 818–833. [5](#), [10](#)
- [29] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, “Deep Residual Learning for Image Recognition,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778. [5](#), [10](#)
- [30] Jeffrey Mahler, Jacky Liang, Sherdil Niyaz, Michael Laskey, Richard Doan, Xinyu Liu, Juan Aparicio Ojea, and Ken Goldberg, “Dex-net 2.0: Deep learning to plan robust grasps with synthetic point clouds and analytic grasp metrics,” *arXiv preprint arXiv:1703.09312*, 2017. [5](#)
- [31] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei, “Deformable convolutional networks,” in *IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 764–773. [7](#)
- [32] Joseph Redmon, “Darknet: Open source neural networks in c,” <http://pjreddie.com/darknet/>, 2013–2016. [10](#)
- [33] Sulabh Kumra and Christopher Kanan, “Robotic grasp detection using deep convolutional neural networks,” in *IEEE International Conference on Intelligent Robots and Systems (IROS)*, 2017, pp. 769–776. [11](#), [13](#), [21](#), [25](#)
- [34] Umar Asif, Jianbin Tang, and Stefan Herrer, “GraspNet: An Efficient Convolutional Neural Network for Real-time Grasp Detection for Low-powered Devices,” in *International Joint Conference on Artificial Intelligence (IJCAI)*, 2018, pp. 4875–4882. [13](#)
- [35] Jeannette Bohg, Antonio Morales, Tamim Asfour, and Danica Kragic, “Data-Driven Grasp Synthesis—A Survey,” *IEEE Transactions on Robotics*, vol. 30, no. 2, pp. 289–309, Mar. 2014. [15](#)
- [36] Ashutosh Saxena, Justin Driemeyer, and Andrew Y Ng, “Robotic grasping of novel objects using vision,” *The International Journal of Robotics Research*, vol. 27, no. 2, pp. 157–173, Feb. 2008. [15](#)
- [37] Yun Jiang, Stephen Moseson, and Ashutosh Saxena, “Efficient grasping from RGBD images: Learning using a new rectangle representation,” in *IEEE International Conference on Robotics and Automation (ICRA)*, 2011, pp. 3304–3311. [15](#), [16](#)
- [38] Edward Johns, Stefan Leutenegger, and Andrew J Davison, “Deep learning a grasp function for grasping under gripper pose uncertainty,” in *IEEE International Conference on Intelligent Robots and Systems (IROS)*. 2016, pp. 4461–4468, IEEE. [15](#)

## REFERENCES

- 
- [39] Jeffrey Mahler, Matthew Matl, Xinyu Liu, Albert Li, David Gealy, and Ken Goldberg, “Dex-Net 3.0: Computing Robust Vacuum Suction Grasp Targets in Point Clouds Using a New Analytic Model and Deep Learning,” in *IEEE International Conference on Robotics and Automation (ICRA)*. May 2018, pp. 5620–5627, IEEE. [15](#)
- [40] Ulrich Viereck, Andreas ten Pas, Kate Saenko, and Robert Platt, “Learning a visuomotor controller for real world robotic grasping using simulated depth images,” in *Conference on Robot Learning (CoRL)*, 2017, pp. 291–300. [15](#)
- [41] Douglas Morrison, Peter Corke, and Jürgen Leitner, “Closing the loop for robotic grasping: A real-time, generative grasp synthesis approach,” *arXiv preprint arXiv:1804.05172*, 2018. [15](#)
- [42] Kourosh Khoshelham and Sander Oude Elberink, “Accuracy and Resolution of Kinect Depth Data for Indoor Mapping Applications,” *Sensors*, vol. 12, no. 2, pp. 1437–1454, Feb. 2012. [15](#)
- [43] Lerrel Pinto and Abhinav Gupta, “Supersizing self-supervision: Learning to grasp from 50K tries and 700 robot hours,” in *IEEE International Conference on Robotics and Automation (ICRA)*. May 2016, pp. 3406–3413, IEEE. [15](#)
- [44] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg, “Ssd: Single shot multibox detector,” in *European conference on computer vision*. Springer, 2016, pp. 21–37. [15](#)
- [45] Xuanchen Zhang, Yuntao Song, Yang Yang, and Hongtao Pan, “Stereo vision based autonomous robot calibration,” *Robotics and Autonomous Systems*, vol. 93, pp. 43–51, 2017. [17](#)
- [46] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie, “Feature pyramid networks for object detection,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2117–2125. [18](#)
- [47] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár, “Focal loss for dense object detection,” in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2980–2988. [21](#)
- [48] Umar Asif, Mohammed Bennamoun, and Ferdous A Sohel, “RGB-D Object Recognition and Grasp Detection Using Hierarchical Cascaded Forests,” *IEEE Transactions on Robotics*, vol. 33, no. 3, pp. 547–564, May 2017. [21](#), [25](#)
- [49] Dongwon Park, Yonghyeok Seo, and Se Young Chun, “Real-time, highly accurate robotic grasp detection using fully convolutional neural network with rotation ensemble module,” in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 9397–9403. [30](#)

---

**REFERENCES**

- [50] Dongwon Park, Yonghyeok Seo, Dongju Shin, Jaesik Choi, and Se Young Chun, “A single multi-task deep neural network with post-processing for object detection with reasoning and robotic grasp detection,” in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 7300–7306. [30](#)

---

# Acknowledgement

---

I cannot believe that it has been already 2 years since I study for master degree. It passed too quickly. Now, I just think what this field is studying and researching, but now I have to finish my degree as a lot of regret. However, I'm going to use this regret as a fuel in the path of a true engineer who applies these technologies in industry, interacting with the world, and uses these technologies to the world advantage a little bit. I think machine learning and deep learning technologies are too important and disruptive tools for engineers who need to come up with answers. I will solve many problems in the world with this methodology with computer vision knowledge, and robot knowledge that interacts with the real environment.

Without so many people around me, this thesis may not have been completed. First of all, I would like to thank to supervisor Professor Se Young Chun for his guidance and teaching for overall of my study and research. If he did not offer me as a master students with this research, I may not achieve this great works and experiences.

I also want to express my sincere thanks to my defense committee members: Professor Sung-Phil Kim, for providing the future works with a researcher's standpoint and encouraging me to understand the level of fusion for multimodal biometrics, and Professor Hwan-Jeong Jeong for his comments about robot experimental issues and results from a robotics perspective.

I would also like to acknowledge lab mates: Hanvit Kim, Dong-Won Park, Thanh Quoc Phan, Mag-aiya Zhussip, Shakarim Soltanayev, Ji-Soo Kim, Kwan-Young Kim, Won-Jae Hong, Byung-Hyun Lee, Ji-ye Son and Ohn Kim. Lastly, I would like to give my very special thanks to my family for always



**REFERENCES**

---

believing in me, especially my parents, who always encouraged me whenever I was down.