2018 IEEE-RAS 18th International Conference on Humanoid Robots (Humanoids)
Beijing, China, November 6-9, 2018

Visual Manipulation Relationship Network for Autonomous Robotics

Hanbo Zhang, Xuguang Lan, Xinwen Zhou, Zhiqiang Tian, Yang Zhang and Nanning Zheng

***Abstract*— robotic grasping is one of the most important fields in robotics, in which great progress has been made in recent years with the help of convolutional neural network (CNN). However, including multiple objects in one scene can invalidate the existing CNN-based grasp detection algorithms, because manipulation relationships among objects are not considered, which are required to guide the robot to grasp things in the right order. This paper presents a new CNN architecture called Visual Manipulation Relationship Network (VMRN) to help robots detect targets and predict the manipulation relationships in real time, which ensures that the robot can complete tasks in a safe and reliable way. To implement end-to-end training and meet real-time requirements in robot tasks, we propose the Object Pairing Pooling Layer (OP^{2}L) to help to predict all manipulation relationships in one forward process. Moreover, in order to train VMRN, we collect a dataset named Visual Manipulation Relationship Dataset (VMRD) consisting of 5185 images with more than 17000 object instances and the manipulation relationships between all possible pairs of objects in every image, which is labeled by the manipulation relationship tree. The experimental results show that the new network architecture can detect objects and predict manipulation relationships simultaneously and meet the real-time requirements in robot tasks.**

I. Introduction

When a robot is interacting with the environment, the first thing is to perceive and understand the environment. For example, when the robot is ordered to get something, it should first make clear what and where the target is, then how to get close to it and grasp it. During interaction with the environment, one of the most important and frequent actions is grasping or manipulation. Therefore, perception and inference before manipulation, which we define as grasp precondition in this paper, is necessary and significant for robots, especially for intelligent robots that may face complex environments containing hundreds of object categories.

As described above, grasping is one of the most significant manipulation in everyday life. robotic grasping has developed rapidly in recent years. However, it is still far behind human performance and remains unsolved. For example, when humans encounter a stack of objects like shown in Fig. 1, they instinctively know how to grasp them. As for the robot, it still remains challenging and, therefore, hinder the widespread use of robots in everyday life.

*This work was supported in part by the key project of Trico-Robot plan of NSFC under grant No. 91748208, National Key Program of China No.2017YFB1302200,key project of Shaanxi province No.2018ZDCXL-GY-06-07, and NSFC No.61573268.

Hanbo Zhang and Xuguang Lan are with the Institute of Artificial Intelligence and Robotics, the National Engineering Laboratory for Visual Information Processing and Applications, School of Electronic and Information Engineering, Xi'an Jiaotong University, No.28 Xianning Road, Xi'an, Shaanxi, China. zhanghanbo163@stu.xjtu.edu.cn, xglan@mail.xjtu.edu.cn

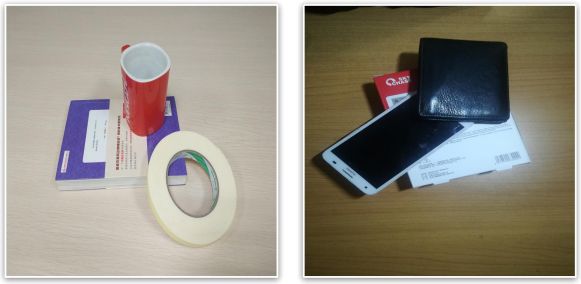


Fig. 1. Importance of manipulation order. Left: A cup is on a book. Right: A phone is on a box. As shown in two scenes, if we do not consider the relationships of manipulation, and robots choose to pick up the book or the box first, then the cup or the phone may be dumped or even broken.

Some recent works have proved the effectiveness of deep learning and convolutional neural network (CNN) in robotic perception [4], [19], [39] and control [8], [18]. In particular, deep learning has achieved unprecedented performance in robotic grasping detection [16], [17], [27], [30]. Most current robotic grasping detection methods take RGB or RGB-D images as input and output vectorized and standardized grasps.

Using this type of grasp detection algorithm for robotic grasping experiments can only deal with scenes containing a single target. Some works try to do grasp or grasp pose detection in cluttered scenes [5], [9], [19], [25], [35]. However, in these works, they just focus on robotic grasping detection or grasp pose estimation without the recognition of what object will be grasped and inference of object relationships that can ensure safe and reliable manipulations. The robot will execute the grasp having the highest confidence score. Doing this can have a devastating effect on objects in some multi-object scenes. For example, as shown in Fig. 1, a cup is placed on a book, and if the detected grasp with the highest confidence score belongs to the book, which means the robot chooses to pick up the book first, the cup may break.

The most relative work that is similar to ours is Guo et al. [10]. In this work, they try to help robots discover the specified target and grasp it. However, this algorithm can only deal with one thing in each iteration and do not concern about the global information. In other words, before grasping the last object, the robot will not know whether the target is in the scene or not. Besides, the original dataset used in this paper only contains 352 RGB-D images and the objects are only fruits, which limits the use in real world tasks.

Therefore, in this paper, we focus on helping the robot infer the right grasping order when it is facing a stack of objects, which is defined as **manipulation relationship**

recognition and will help robots manipulate or grasp things in a safe mode.

Some recent works have used CNNs to predict the relationships between objects rather than just object detection [1], [20], [23], [36]. These works show that the CNN has the potential to understand the relationships between objects. Therefore, we hope to establish a method based on neural network so that the robot can understand the manipulation relationships between objects in multi-object scenes to help the robot finish more complicated grasping tasks.

In our work, we design a new network architecture named Visual Manipulation Relationship Network (VMRN) to simultaneously detect objects and recognize the manipulation relationships. The network architecture has two stages. The output of the first-stage is the object detection result, and the output of the second-stage is the recognition result of the manipulation relationships. To train our network, we contribute a new dataset called Visual Manipulation Relationship Dataset (VMRD). The dataset contains 5185 images of hundreds of objects with 51530 manipulation relationships and the category and location information of each object. In summary, the contributions of our work include three points:

- We design a new CNN architecture to simultaneously detect objects and recognize manipulation relationships, which meets the real-time requirements in robot tasks.
- We collect a new dataset of hundreds graspable objects, which includes the location and category information and the manipulation relationships between pairs of objects.
- As we know, it is the first end-to-end architecture to predict robotic manipulation relationships directly using an image as input with a CNN.

II. Background

A. Object Detection

Object detection is defined as a process using an image including several objects as input to locate and classify as many target objects as possible in the image. Sliding window used to be the most common method to detect objects. When using this way to do object detection, the features, such as HOG [2] and SIFT [22], of the target objects are usually extracted first, and then they are used to train a classifier, like Supported Vector Machine, to classify the candidates coming from sliding window stage. Deformable Parts Model (DPM) [3] is the most successful one of this type of object detection algorithms.

Recently, object detection algorithms based on deep features, such as Region-based CNN (RCNN) family [31], [33] and Single Shot Detector (SSD) family [21], are proved to drastically outperform the previous algorithms which are based on hand-designed features. Based on the detection process, the main algorithms are classified into two types, which we call one-stage algorithms such as SSD [21] and two-stage algorithms such as Faster RCNN [31]. One-stage algorithms are usually faster than two-stage algorithms while two-stage algorithms often get better results [12].

Our work focuses on not only the object detection, but also the manipulation relationship recognition. The challenge is how to combine the relationship recognition stage with object detection stage. To solve this problem, we design the Object Pairing Pooling Layer, which is used to generate the input of manipulation relationship predictor using the object detection results and convolutional feature maps as input. The details will be described in following sections.

B. Visual Relationship Detection

Visual relationship detection means understanding object relationships of an image. Some previous works try to learn spatial relationships [6], [7]. Later, researchers attempt to collect relationships from images and videos and help models map these relationships from images to language [15], [29], [34], [41]. Recently, with the help of deep learning, the relationship recognition between objects has made a great process [1], [20], [23], [36]. Lu et al. [23] collect a new dataset of object relationships called Visual Relationship Dataset and propose a new relationship recognition model consisting of visual and language parts, which outperforms previous methods. Liang et al. [20] firstly combine deep reinforcement learning with relationships and their model can sequentially output the relationships between objects in one image. Yu et al. [36] use internal and external linguistic knowledge to compute the conditional probability distribution of a predicate given a $(subject, object)$ pair, which achieves better performance. Dai et al. [1] propose an integrated framework called Deep Relational Network for exploiting the statistical dependencies between objects and their relationships.

These works focus on relationships represented by linguistic information between objects but not manipulation relationships. In our work, we introduce relationship detection methods to help robots find the right order in which the objects should be manipulated. And because of the real-time requirements of robot system, we need to find a way to accelerate the recognition of manipulation relationships. Therefore, we propose an end-to-end architecture different from all previous works.

C. Spatial Relationship Reasoning

Spatial relationship in robotics is similar to our manipulation relationship. Spatial relationship reasoning often takes point clouds of the environment as input, and through analysis, gets spatial relationships between objects in the scene. Rosman et al. [32] segment objects in the point cloud, use SVM to extract contact point network for spatial relationship reasoning and redescribe a scene in terms of a layered representation to help robots manipulate interacting objects in a meaningful way. Zampogiannis et al. [37] use point cloud tracking to equip robot the ability of understanding and reproducing the evolution of the spatial relations between involved objects during observing and executing complex manipulation actions such as pouring water and placing objects in a bowl. Ziaeetabar et al. [40] apply spatial relationship reasoning in semantically comparing and

identifying actions, which is called Enriched Semantic Event Chain representation. Another term "support relationship" is similar to spatial relationship, which means the support order of stacked objects [24]. In [24], geometry and static equilibrium in classical mechanics are used to support relations between object pairs. Later, single and multiple view support order is inferred into three classes: "support from below", "support from side", and "containment" [26]. Recently, a safe manipulation strategy based on spatial relationship is proposed in [14], which can handle situations uncertainties in support relation.

However, spatial relationship differs from manipulation relationship since it focuses on the relative position instead of directly manipulation order. Our proposed manipulation relationship network directly outputs which object should be manipulated and grasped first to ensure the stability of the other objects. It is more simple and only needs RGB images instead of point clouds to do the inference, which is inspired by the idea of visual relationship detection.

III. Proposed Approach

The proposed network architecture is shown in Fig. 3. The inputs of our network are images and outputs are object detection results and manipulation relationship trees. Our network consists of three parts: feature extractor, object detector and manipulation relationship predictor, with parameters denoted by Φ, Ω and Θ respectively.

In our work, taking into account the real-time requirements of object detection, we use Single Shot Detector (SSD) algorithm [21] as our object detector. SSD is an one-stage object detection algorithm based on CNN. It utilizes multi-scale feature maps to regress and classify bounding boxes in order to adapt to object instances with different size. Input of object detector is convolution feature maps (in our work, we use VGG16 [13] or ResNet50 [11] features). Through object classification and multi-scale object location regression, we obtain the final object detection results. The result of each object is a 5-dimensional vector $(cls, x_{min}, y_{min}, x_{max}, y_{max})$. Then the inputs of Object Pairing Pooling Layer (OP^{2}L) are object detection results and convolution features. The outputs are concatenated as a mini-batch for predicting manipulation relationships by traversing all possible pairs of objects. Finally, the manipulation relationship between each pair of objects is predicted by manipulation relationship predictor.

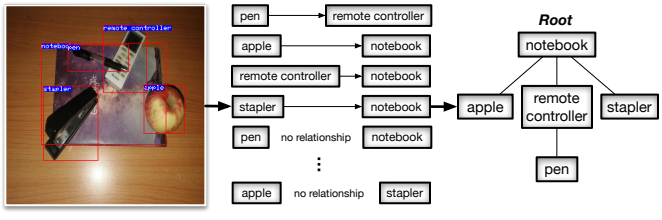


Fig. 2. An example of manipulation relationship tree. Left: Images including several objects. Middle: All pair of objects and manipulation relationships. Right: manipulation relationship tree, in which the leaf nodes should be manipulated before the other nodes.

A. Manipulation Relationship Representation

In this paper, manipulation relationship is the order of grasping. Therefore, we need a objective criterion to determine the grasping order, which is described as following: if moving one object will have an effect on the stability of another object, this object should not be moved first. Since we only focus on the manipulation relationships between objects and do not concern the linguistic information, a tree-like structure (two objects may have one same child), called **manipulation relationship tree** in the following, can be constructed to represent the manipulation relationships of all the objects in each image. Objects are represented by nodes and parent-child relationships between nodes indicate the manipulation relationships. In manipulation relationship tree, the object represented by the parent node should be grasped after the object represented by the child node. Fig. 2 is an example of the manipulation relationship tree. A pen is on a remote controller, and a remote controller, an apple and a stapler are on a book. Therefore, the pen is the child of the remote controller and the remote controller, the apple and the stapler are children of the book in the manipulation tree.

B. Object Pairing Pooling Layer

OP^{2}L is designed to implement the end-to-end training of the whole network. In our work, weights of feature extractor Φ are shared by manipulation relationship predictor and object detector. OP^{2}L is added between feature extractor and manipulation relationship predictor like in Fig. 3, using object location ($e.g.$ the online output of object detection or the offline ground truth bounding box) and shared feature maps $CNN(I;\Phi)$ as input, where I is the input image. It finds out all possible pairs (n objects correspond to $n(n-1)$ pairs) of objects and makes their features a mini-batch to train the manipulation relationship predictor. Although in complex visual relationship recognition tasks, traversing all possible object pairs is time-consuming [38] due to the large number of objects in the scene and the sparsity of the relationships between the objects. However, in our manipulation relationship recognition task, there are only a few of objects in the scene and it does not take a long time to traverse all the object pairs.

Let O_i and O_j stand for an object pair. OP^{2}L can generate the features of O_i and O_j denoted by $CNN(O_i, O_j;\Phi)$, which includes features of two objects and their union. In detail, the features are cropped from shared feature maps and adaptively pooled into a fixed spatial size $H \times W$ ($e.g. 7 \times 7$). The gradients with respect to the same object or union bounding box coming from manipulation relationship predictor are accumulated and propagated backward to the front layers.

C. Training Data of Relation Predictor

An extra branch of CNN is cascaded after OP^{2}L to predict manipulation relationships between objects. Training data for manipulation relationship predictor D_{RP} is generated by OP^{2}L, which includes two parts: online data

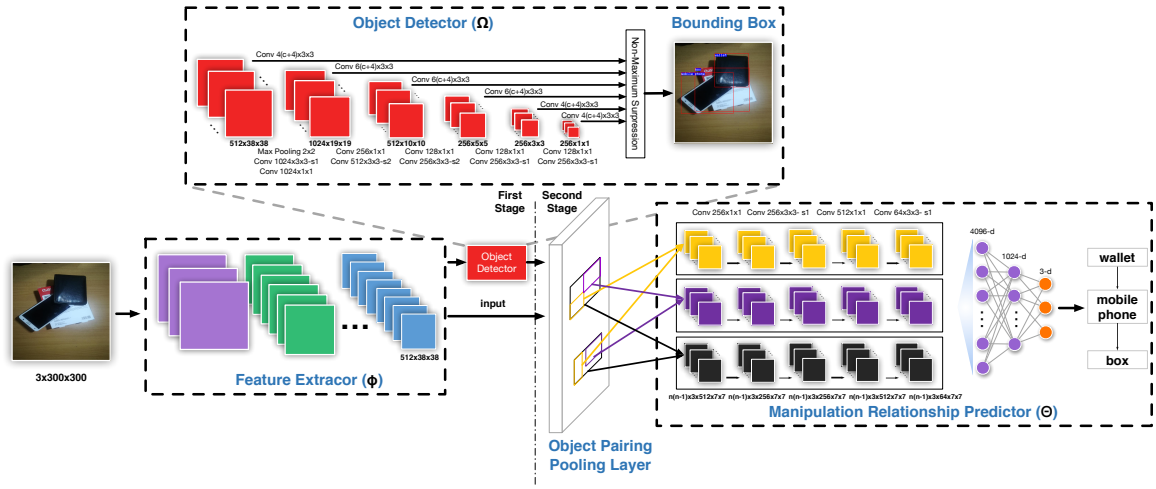


Fig. 3. Network architecture of VMRN. Input of the network is $3 \times 300 \times 300$ images including several graspable objects. Feature extractor is a stack of convolution layers ($e.g.$ VGG [13] or ResNet [11]), which output feature maps with size of $512 \times 38 \times 38$. These features are used by object detector and OP^{2}L to respectively detect objects and generate the feature groups of all possible object pairs which are used to predict manipulation relationships by manipulation relationship predictor.

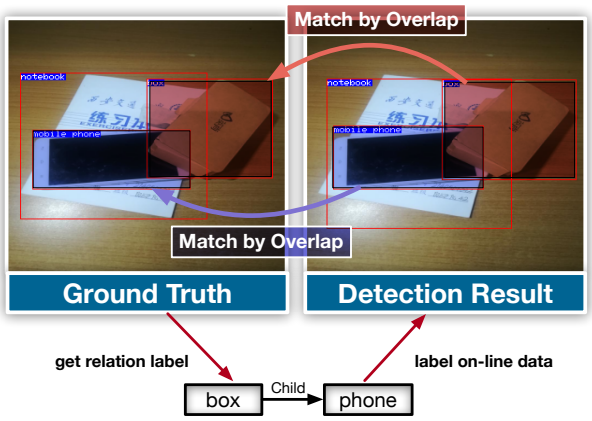


Fig. 4. Method to label online data. First, we match the predicted bounding boxes to the ground truth by areas of overlap. Then we use the manipulation relationship between ground truth bounding boxes as the ground truth manipulation relationship between predicted bounding boxes to generate online data used to train manipulation relationship predictor.

D_{on} and offline data D_{off}, coming from object detection results and ground truth bounding boxes respectively. That is to say $D_{RP} = D_{on} \cup D_{off}$. For each image, D_{RP} is a set of CNN features $CNN(O_i, O_j; \Phi)$ of all possible object pairs and their labels (O_i, R, O_j), where R is the manipulation relationship between O_i and O_j. The reason we mix online data and offline data to train manipulation relationship predictor is that online data can be seen as the augmentation of offline data while offline data can be seen as the correction of online data. Manipulation relationships between online object instances are labeled according to the manipulation relationships between ground truth bounding boxes that maximumly overlap the online ones. As shown in Fig. 4, object detection result is shown in right. The manipulation relationship between the mobile phone and the box is determined by the following two steps: 1) match detected bounding boxes of the mobile phone and the box to the ground truth ones by overlaps; 2) use manipulation relationship between the ground truth bounding boxes to label the manipulation relationship of detected bounding boxes.

D. Loss Function of Relation Predictor

In our work, there are three manipulation relationship types between any two objects in one image:

- 1) object 1 is the parent of object 2
- 2) object 2 is the parent of object 1
- 3) object 1 and object 2 have no manipulation relationship.

Therefore, our manipulation relationship recognition process is essentially a classification problem of three categories for any pair of objects $CNN(O_i, O_j; \Phi)$. Let Θ denote the weights of relation recognition branch. Note that because exchanging the subject and object will possibly change the manipulation relationship type ($e.g.$ from $parent$ to $child$), the resultsrecognition of $CNN(O_i, O_j; \Phi)$ and $CNN(O_j, O_i; \Phi)$ may be different. The manipulation relationship likelihood of R is defined as:

$$P(R|O_i, O_j; \Theta) = \frac{e^{h^R_\Theta(CNN(O_i,O_j;\Phi))}}{\sum_{i=1}^{3} e^{h^i_\Theta(CNN(O_i,O_j;\Phi))}} \quad (1)$$

We choose multi-class cross entropy function as loss function of manipulation relationship recognition:

$$L_{rp}(R|O_i, O_j; \Theta) = -log(P(R|O_i, O_j; \Theta)) \quad (2)$$

For each image, manipulation relationship recognition loss includes two parts: online data loss L_{on} and offline data loss L_{off}. The loss for the whole image is:

$$
\begin{aligned}
L_{RP}(D_{RP};\Theta) =& \lambda L_{on} + (1-\lambda)L_{off} \\
=& \lambda \sum_{D_{on}} L_{rp}(R|O_i,O_j;\Theta) + \\
& (1-\lambda) \sum_{D_{off}} L_{rp}(R|O_i,O_j;\Theta)
\end{aligned} \tag{3}
$$

where λ is used to balance the importance of online data D_{on} and offline data D_{off}. In our work, we set λ to 0.5.

E. Training Method

The whole network is trained end-to-end, which means that the object detector and manipulation relationship predictor are trained simultaneously.

Let Ω be the weights of object detector and D_{OD} be the training data of object detector including shared features of the whole image $CNN(I;\Phi)$ and object detection ground truth $(\overline{cls},\overline{loc})$. The loss function for object detector is the same as Liu et al. described in [21]:

$$
L_{OD}(D_{OD};\Omega) = L_{loc} + \alpha L_{conf} \tag{4}
$$

where α is set to 1 according to experience. Like in [21], $default\ bounding\ boxes$ are defined as a set of predetermined bounding boxes with a few of fixed sizes, which serve as a reference during object detection process. Location loss L_{loc} is smooth L1 loss between ground truth bounding box and matched default bounding box and all bounding boxes are encoded as offsets. Classification confidence loss L_{conf} is also multi-class cross entropy loss.

Loss function of manipulation relationship recognition L_{RP} is detailed in section IV.B. Combining L_{RP} and L_{OD}, the complete loss for shared layers is:

$$
L(I;\Phi) = \mu L_{OD}(D_{OD};\Omega) + (1-\mu)L_{RP}(D_{RP};\Theta) \tag{5}
$$

μ is used to balance the importance of L_{OD} and L_{RP}. In our work, μ is set to 0.5. And according to chain rule:

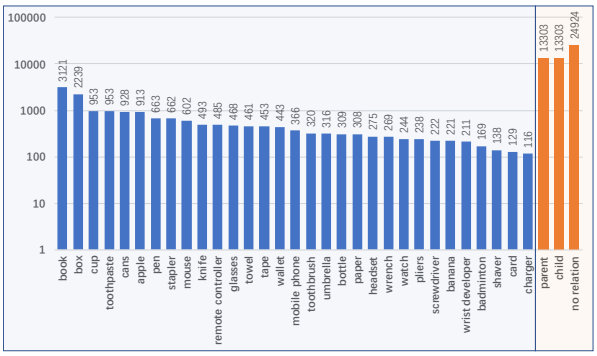
$$
\begin{aligned}
\frac{\partial L}{\partial \Phi} =& \mu \frac{\partial L_{OD}}{\partial CNN(I;\Phi)} \frac{\partial CNN(I;\Phi)}{\partial \Phi} + \\
& (1-\mu) \frac{\partial L_{RP}}{\partial CNN(O_i,O_j;\Phi)} \frac{\partial CNN(O_i,O_j;\Phi)}{\partial \Phi}
\end{aligned} \tag{6}
$$

IV. Dataset

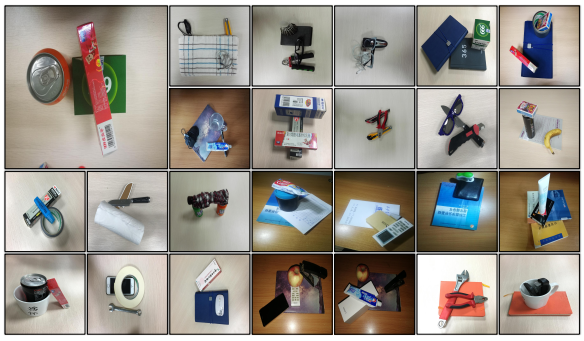
A. Data Collection

Different from visual relationship dataset [23], we focus on manipulation relationships, so objects included in our dataset should be manipulatable or graspable. Moreover, manipulation relationship dataset should contain not only objects localized in images, but also rich variety of position relationships.

Our data are collected and labeled using hundreds of objects coming from 31 categories. There are totally 5185 images including 17688 object instances and 51530 manipulation relationships. Category and manipulation relationship distribution is shown in Fig. 5(a). Each object node includes



(a) Category and manipulation relationship distrubution



(b) Dataset examples

Fig. 5. Visual Manipulation Relationship Dataset. (a) Category and manipulation relationship distribution of our dataset. (b) Some dataset examples

category information, bounding box location, the index of the current node and indexes of its parent nodes and child nodes. Some examples of our dataset is shown in Fig. 5(b). Complete dataset can be downloaded at this link[1].

During training, we randomly split the dataset into a training set and a testing set in a ratio of nine to one. In detail, training set includes 4656 images, 15911 object instances and 46934 manipulation relationships, and testing set contains the rest.

B. Labeling Criterion

Because our dataset focuses on the manipulation relationship with no linguistic or position information, instead of directly giving position relationships ($e.g.$ under, on, beside and so on) between objects, we only give the order of manipulation of objects: manipulation relationship tree. There are several advantages over giving position relationships: 1) the manipulation relationships are more simpler, which makes relationship recognition task easier; 2) the results can directly give the manipulation relationships between objects, without the need to reconstruct the manipulation relationships through position relationships.

During labeling, there should be a criterion that can be strictly enforced. Therefore, in our work, we set a **labeling criterion** of manipulation relationship: when the movement

[1]http://gr.xjtu.edu.cn/web/zeuslan/visual-manipulation-relationship-dataset;jsessionid=E6F2B81979FA7849D9D773A7A389B809

TABLE I

ACCURACY OF OBJECT DETECTION AND VISUAL MANIPULATION RELATIONSHIP RECOGNITION

Author	Algorithm	Training Data	mAP	Rel.	Obj.Rec.	Obj.Prec.	Img.	Speed (ms)
Lu et al. [23]	VGG16-SSD, VAM	$D_{on} \cup D_{off}$	93.01	88.76	75.50	71.28	46.88	>100
	ResNet50-SSD, VAM	$D_{on} \cup D_{off}$	91.72	88.76	74.33	75.19	49.72	
Ours	VGG16-VMRN (No Rel. Grad.)	$D_{on} \cup D_{off}$	93.01	88.36	77.28	73.04	50.66	**28**
	ResNet50-VMRN (No Rel. Grad.)	$D_{on} \cup D_{off}$	91.72	90.73	77.68	77.55	53.12	
	VGG16-VMRN	D_{on}	94.18	92.80	**82.64**	77.76	60.49	
	VGG16-VMRN	D_{off}	**94.36**	92.75	81.55	76.09	58.60	
	VGG16-VMRN	$D_{on} \cup D_{off}$	94.09	**93.36**	82.29	**78.01**	**63.14**	
	ResNet50-VMRN	D_{on}	91.81	92.01	79.03	72.55	54.44	
	ResNet50-VMRN	D_{off}	92.71	91.86	79.33	74.71	55.95	
	ResNet50-VMRN	$D_{on} \cup D_{off}$	92.67	92.19	80.55	76.02	57.28	

of an object will affect the stability of other objects, the object should not be the leaf node of the manipulation relationship tree, which means that the object should not be moved first. For example, as shown in the up-left image in Fig. 5(b), there are three objects: on the left, there is an orange can and on the right, a red box is put on a green box. If the green box is moved first, it will have an effect on the stability of the red box, so it should not be the leaf node of the manipulation relationship tree. If the red box or the can is moved first, it will not affect stability of any other object, so they should be the leaf node.

V. EXPERIMENTS

A. Training Settings

Our models are trained on Titan Xp with 12 GB memory. We have trained two Visual Manipulation Relationship Network (VMRN) models based on VGG16 net and ResNet50 called VGG16-VMRN and ResNet50-VMRN. Because of the unstability of the random object detection results in the beginning, the two VMRN models are pretrained with only D_{off} for the first 10k iterations. Learning rate for both networks is 0.001 and will decay to 0.0001 after 80k iterations. Weight decay for VGG16 and ResNet is 0.003 and 0.0001 respectively. Batch size is 8 and momentum is 0.9. We set Nesterov to True for both networks. Training will take 120 epochs with 1000 iterations per epoch. The hyperparameters are shared with object detector and manipulation relationship predictor.

B. Testing Settings

Comparison Model As we know, there is no research about vision-based robotic manipulation relationship recognition with CNN so far. Therefore, we compare our experimental results with Visual Appearance Model (VAM) in Lu et al. [23], which is modified to adapt to our task. VAM takes union bounding box as input and outputs the relationship. But in our work, exchanging the subject and object may change the manipulation relationship. Therefore, instead of only using union bounding box, we parallel subject, object and union bounding boxes as input to get the final manipulation relationship.

Self Comparison To study the contribution of OP^{2}L and end-to-end training, we also confirm the performance of our models that are trained with no gradients backward from manipulation relationship predictor ({VGG16-SSD, VMRN (No Rel. Grad.)} and {ResNet50-SSD, VMRN (No Rel. Grad.)}). To explore the benefits from online and offline data, we also train our models with only online (D_{on}) or offline (D_{off}) training data.

Metrics Three metrics are used in our experiment: 1) Manipulation Relationship Testing (Rel.): this metric focus on the accuracy of manipulation relationship model on ground truth object instance pairs, in which the input features or image patches of manipulation relationship predictor are obtained based on the offline ground truth bounding boxes; 2) Object-based Testing (Obj. Rec. and Obj. Prec.): this metric tests the accuracy based on object pairs. In this setting, the triplet (O_i, R, O_j) is treated as a whole. The result is considered correct if both objects are detected correctly (category is right and IoU between predicted bounding box and ground truth is more than 0.5) and the predicted manipulation relationship is correct. We compute the recall (Obj. Rec.) and precision (Obj. Prec.) of our models during object-based testing 3) Image-based Testing (Img.): this metric tests the accuracy based on the whole image. In this setting, the image is considered correct only when all possible triplets are predicted correctly.

C. Analysis

Results are shown in Table I. Compared with VAM, we can conclude that:

1) **Performance is better:** VAM performs worse than proposed VMRN models in all three experiment settings. The gains mainly come from the end-to-end training process, which improves the accuracy of manipulation relationship a lot (from 88.76% to 93.36%). This is confirmed in the following self comparison part.

2) **Speed is faster:** The proposed VMRN models (VGG-VMRN and ResNet-VMRN) are both less time-consuming than VAM. Forward process of OP^{2}L and manipulation relationship predictor takes 5.5ms per image in average. As described in [21], the speed of SSD object detector

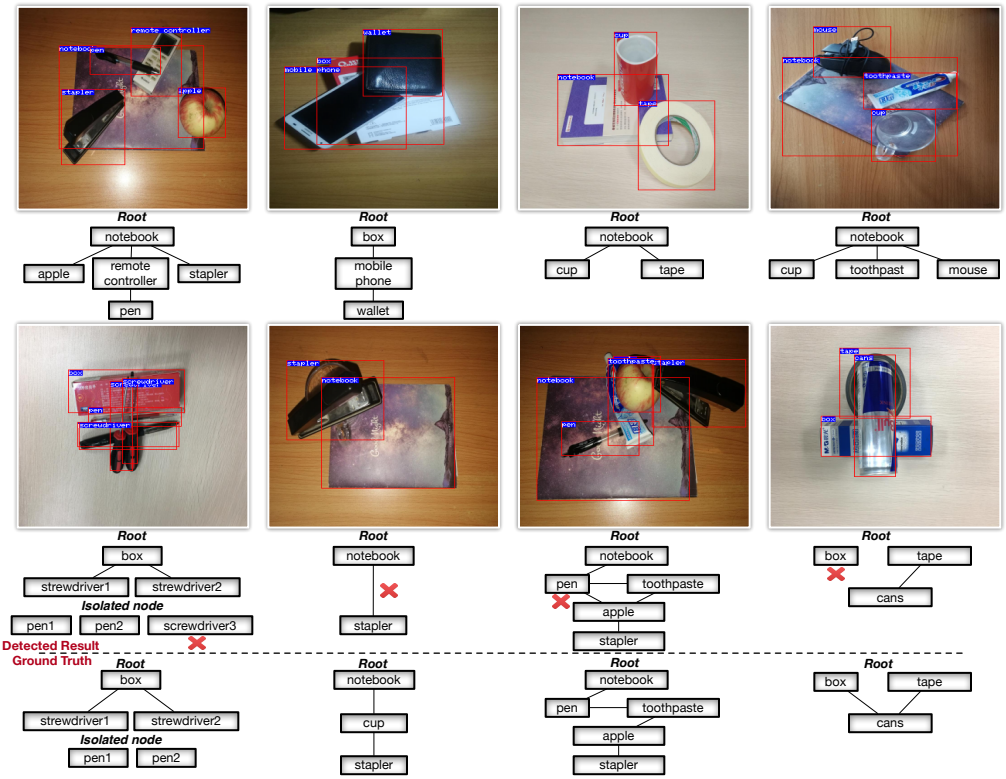


Fig. 6. Result examples. Upper: examples with right object detection and manipulation relationship . Lower: examples with wrong results (from left to right: redundant object detection, failing object detection, redundant manipulation relationship, failing manipulation relationship)

is 21.74ms per image on Titan X with mini-batch of 1 image. Therefore, our manipulation relationship recognition has little effect on speed of the whole network. But because of the huge network architecture and sequential process, VAM spends 122ms on each image in average to predict all of the manipulation relationships. Even when we put all possible triplets of one image to a batch, it still spends 86ms for one image.

Self comparison results indicate that proposed VMRN models trained end-to-end can outperform the models that are trained without the gradients from manipulation relationship predictor. It mainly benefits from the influence coming from manipulation relationship recognition loss L_{RP}. The parameters of the network are adjusted to better predict the visual manipulation relationships and the network is more holistic. As explored in Pinto et al. [28], multi-task learning in our network can help improve the performance because of diversity of data and regularization in learning. Finally, we can observe that using online and offline data simultaneously may actually help to improve the performance of the network due to the complementing of online and offline data.

The difference between the performance of VGG16-VMRN and ResNet50-VMRN is also interesting. Gradients coming from manipulation relationship recognition loss L_{RP} improve both networks, but its improvement on ResNet50-VMRN is less than that on VGG16-VMRN as shown in Table I. Note that VGG16-based feature extractor has 7.63 million parameters and ResNet50-based feature extractor has 1.45 million parameters, so the number of parameters may limit the performance ceiling of ResNet50-VMRN. In the future, we will try deeper ResNet as our base network.

Some subjective results are shown in Fig. 6. From the four examples in the first line, we can see that our model can simultaneously detect objects and manipulation relationships in one image. From the four examples in the second line, we can conclude that the occlusion, the similarity between different categories and visual illusion can have a negative influence on the predicted results.

Note that when there are more than one objects of the same kind in one image, VMRN will also work well because objects are distinguished by the indexes corresponding to them, which are bound to them when they are detected.

VI. Conclusions

In this paper, we focus on solving the problem of visual manipulation relationship recognition to help robots manipulate things in the right order. We propose a new network architecture named Visual Manipulation Relationship Network and collect a dataset called Visual Manipulation Relationship Dataset to implement simultaneously object detection and manipulation relationship recognition, which meets the real-time requirement on robot platform. The proposed Object Paring Pooling Layer (OP^2L) can not only accelerate the manipulation relationship recognition by replacing the sequential process with a simple forward process, but also improve the performance of the whole network by back-

propagating the gradients from manipulation relationship predictor.

However, due to the limited number of objects used in training, it is difficult for the object detector to generalize to objects with a large difference in appearance from our dataset. Besides, due to the traversal of all object pairs in the scene, when there are too many objects, memory usage of the network will become unacceptable. In our future work, we will expand our dataset using more graspable objects and combine the grasp detection with VMRN to implement an all-in-one network which can simultaneously detects objects and their grasp positions and recognizes the correct manipulation relationships. Moreover, we will try to overcome the memory usage shortcoming for scenes with large number of objects.

ACKNOWLEDGMENT

This work was supported in part by the key project of Trico-Robot plan of NSFC under grant No. 91748208, National Key Program of China No.2017YFB1302200,key project of Shaanxi province No.2018ZDCXL-GY-06-07, and NSFC No.61573268. Thanks for the high-quality advices by the revierwers for improving this paper.

REFERENCES

[1] B. Dai, Y. Zhang, and D. Lin. Detecting visual relationships with deep relational networks. *arXiv preprint arXiv:1704.03114*, 2017.

[2] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, volume 1, pages 886–893. IEEE, 2005.

[3] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1627–1645, 2010.

[4] C. Finn, X. Y. Tan, Y. Duan, T. Darrell, S. Levine, and P. Abbeel. Deep spatial autoencoders for visuomotor learning. In *ICRA*, pages 512–519. IEEE, 2016.

[5] D. Fischinger, M. Vincze, and Y. Jiang. Learning grasps for unknown objects in cluttered scenes. In *ICRA*, pages 609–616. IEEE, 2013.

[6] C. Galleguillos, A. Rabinovich, and S. Belongie. Object categorization using co-occurrence, location and appearance. In *CVPR*, pages 1–8. IEEE, 2008.

[7] S. Gould, J. Rodgers, D. Cohen, G. Elidan, and D. Koller. Multi-class segmentation with relative location prior. *International Journal of Computer Vision*, 80(3):300–316, 2008.

[8] S. Gu, E. Holly, T. Lillicrap, and S. Levine. Deep reinforcement learning for robotic manipulation with asynchronous off-policy updates. In *ICRA*, pages 3389–3396. IEEE, 2017.

[9] M. Gualtieri, A. ten Pas, K. Saenko, and R. Platt. High precision grasp pose detection in dense clutter. In *IROS*, pages 598–605. IEEE, 2016.

[10] D. Guo, T. Kong, F. Sun, and H. Liu. Object discovery and grasp detection with a shared convolutional neural network. In *ICRA*, pages 2038–2043. IEEE, 2016.

[11] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016.

[12] J. Huang, V. Rathod, C. Sun, M. Zhu, A. Korattikara, A. Fathi, I. Fischer, Z. Wojna, Y. Song, S. Guadarrama, et al. Speed/accuracy trade-offs for modern convolutional object detectors. In *CVPR*, 2017.

[13] S. Karen and Z. Andrew. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[14] R. Kartmann, F. Paus, M. Grotz, and T. Asfour. Extraction of physically plausible support relations to predict and validate manipulation action effects. *IEEE Robotics and Automation Letters*, 3(4):3991–3998, 2018.

[15] G. Kulkarni, V. Premraj, V. Ordonez, S. Dhar, S. Li, Y. Choi, A. C. Berg, and T. L. Berg. Babytalk: Understanding and generating simple image descriptions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(12):2891–2903, 2013.

[16] S. Kumra and C. Kanan. Robotic grasp detection using deep convolutional neural networks. In *IROS*. IEEE, 2017.

[17] I. Lenz, H. Lee, and A. Saxena. Deep learning for detecting robotic grasps. *The International Journal of Robotics Research*, 34(4-5):705–724, 2015.

[18] S. Levine, C. Finn, T. Darrell, and P. Abbeel. End-to-end training of deep visuomotor policies. *Journal of Machine Learning Research*, 17(39):1–40, 2016.

[19] S. Levine, P. Pastor, A. Krizhevsky, J. Ibarz, and D. Quillen. Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection. *The International Journal of Robotics Research*, 2016.

[20] X. Liang, L. Lee, and E. P. Xing. Deep variation-structured reinforcement learning for visual relationship and attribute detection. In *CVPR*, pages 4408–4417. IEEE, 2017.

[21] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg. Ssd: Single shot multibox detector. In *ECCV*, pages 21–37. Springer, 2016.

[22] D. G. Lowe. Object recognition from local scale-invariant features. In *ICCV*, volume 2, pages 1150–1157. Ieee, 1999.

[23] C. Lu, R. Krishna, M. Bernstein, and L. Fei-Fei. Visual relationship detection with language priors. In *ECCV*, pages 852–869. Springer, 2016.

[24] R. Mojtahedzadeh, A. Bouguerra, E. Schaffernicht, and A. J. Lilienthal. Support relation analysis and decision making for safe robotic manipulation tasks. *Robotics and Autonomous Systems*, 71:99–117, 2015.

[25] P. Ni, W. Zhang, W. Bai, M. Lin, and Q. Cao. A new approach based on two-stream cnns for novel objects grasping in clutter. *Journal of Intelligent and Robotic Systems*, pages 1–17, 2018.

[26] S. Panda, A. A. Hafez, and C. Jawahar. Single and multiple view support order prediction in clutter for manipulation. *Journal of Intelligent & Robotic Systems*, 83(2):179–203, 2016.

[27] L. Pinto and A. Gupta. Supersizing self-supervision: Learning to grasp from 50k tries and 700 robot hours. In *ICRA*, pages 3406–3413. IEEE, 2016.

[28] L. Pinto and A. Gupta. Learning to push by grasping: Using multiple tasks for effective learning. In *ICRA*, pages 2161–2168. IEEE, 2017.

[29] V. Ramanathan, C. Li, J. Deng, W. Han, Z. Li, K. Gu, Y. Song, S. Bengio, C. Rosenberg, and L. Fei-Fei. Learning semantic relationships for better action retrieval in images. In *CVPR*, pages 1100–1109, 2015.

[30] J. Redmon and A. Angelova. Real-time grasp detection using convolutional neural networks. In *ICRA*, pages 1316–1322. IEEE, 2015.

[31] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NIPS*, pages 91–99, 2015.

[32] B. Rosman and S. Ramamoorthy. Learning spatial relationships between objects. *The International Journal of Robotics Research*, 30(11):1328–1342, 2011.

[33] G. Ross. Fast r-cnn. In *ICCV*, pages 1440–1448. IEEE, 2015.

[34] J. Thomason, S. Venugopalan, S. Guadarrama, K. Saenko, and R. J. Mooney. Integrating language and vision to generate natural language descriptions of videos in the wild. In *COLING*, volume 2, page 9, 2014.

[35] U. Viereck, A. t. Pas, K. Saenko, and R. Platt. Learning a visuomotor controller for real world robotic grasping using simulated depth images. *arXiv preprint arXiv:1706.04652*, 2017.

[36] R. Yu, A. Li, V. I. Morariu, and L. S. Davis. Visual relationship detection with internal and external linguistic knowledge distillation. In *CVPR*, pages 1974–1982, 2017.

[37] K. Zampogiannis, Y. Yang, C. Fermüller, and Y. Aloimonos. Learning the spatial semantics of manipulation actions through preposition grounding. In *ICRA*, pages 1389–1396. IEEE, 2015.

[38] J. Zhang, M. Elhoseiny, S. Cohen, W. Chang, and A. Elgammal. Relationship proposal networks. In *CVPR*, volume 1, page 2, 2017.

[39] X. Zhou, X. Lan, H. Zhang, Z. Tian, Y. Zhang, and N. Zheng. Fully convolutional grasp detection network with oriented anchor box. In *IROS*, 2018.

[40] F. Ziaeetabar, E. E. Aksoy, F. Wörgötter, and M. Tamosiunaite. Semantic analysis of manipulation actions using spatial relations. In *ICRA*, pages 4612–4619. IEEE, 2017.

[41] C. L. Zitnick, D. Parikh, and L. Vanderwende. Learning the visual interpretation of sentences. In *ICCV*, pages 1681–1688, 2013.