2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
Macau, China, November 4-8, 2019

Task-oriented Grasping in Object Stacking Scenes with CRF-based Semantic Model

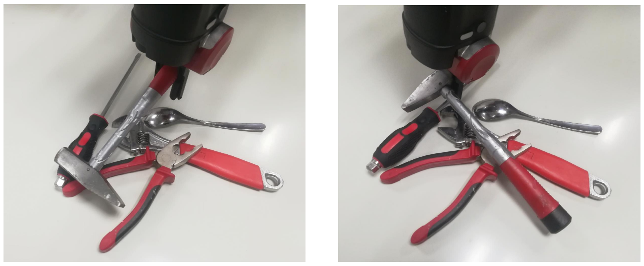
Chenjie Yang, Xuguang Lan, Hanbo Zhang and Nanning Zheng

***Abstract*— In task-oriented grasping, the robot is supposed to manipulate the objects in a task-compatible manner, which is more important but more challenging than just stably grasping. However, most of existing works perform task-oriented grasping only in single object scenes. This greatly limits their practical application in real world scenes, in which there are usually multiple stacked objects with serious overlaps and occlusions. To perform task-oriented grasping in object stacking scenes, in this paper, we firstly build a synthetic dataset named Object Stacking Grasping Dataset (OSGD) for task-oriented grasping in object stacking scenes. Secondly, a Conditional Random Field (CRF) is constructed to model the semantic contents in object regions. The modelled semantic contents can be illustrated as incompatibility of task labels and continuity of task regions. This proposed approach can greatly reduce the interference of overlaps and occlusions in object stacking scenes. To embed the CRF-based semantic model into our grasp detection network, we implement the inference process of CRFs as a RNN so that the whole model, Task-oriented Grasping CRFs (TOG-CRFs) can be trained end to end. Finally, in object stacking scenes, the constructed model can help robot achieve 69.4% success rate for task-oriented grasping.**

I. INTRODUCTION

In tool manipulation, robot is supposed to choose a grasp pose which satisfies not only the stability requirements, but the task constraints. For example, considering a hammer, the robot is supposed to hold its handle when performing the task hammering but leave the handle clear when delivering it to others, which is shown in Fig. 1. Due to its significance, task-oriented grasping has been studied deeply in many fields, such as robot mechanics [1]–[4] and computer vision [5]–[8]. However, almost all of these works only detect grasps in single object scenes, while the real-world scenes often contain multiple objects. And more seriously, the objects may often be stacked together as in Fig. 1, which immensely increases the difficulty of task-oriented grasping.

Algorithms based on robot mechanics may generate more stable and accurate task-compatible grasps. However, these algorithms usually require the full 3D models of objects, which limits their application in unstructured environments. Meanwhile, with the development of deep learning [9]–[12], the vision-based grasping algorithms can also achieve satisfactory results. More importantly, these algorithms can be easily generalized to real-world scenes because of the availability of visual information. However, when there are multiple stacked objects in scenes, this kind of grasping algorithms will be severely degraded due to the serious overlaps and occlusions.



(a) Hammer (b) Hand Over

Fig. 1. The illustration of task-oriented grasping in object stacking scenes. (a) When performing the task hammering, we are supposed to firstly choose a suitable tool from these stacked objects, such as a hammer, and then grasp it away from its heavier end. (b) If we want to hand over the hammer, we are supposed to grasp its head or neck while leaving its handle clear.

To perform task-oriented grasping in object stacking scenes, in this paper, we firstly construct a synthetic dataset named Object Stacking Grasping Dataset (OSGD). Then, to reduce the interference caused by overlaps and occlusions, we build a Conditional Random Field (CRF) based semantic model and embed it into our grasp detection network. The whole model will be named as Task-oriented Grasping CRFs (TOG-CRFs). In this framework, we firstly use Region Proposal Network (RPN) [13] to generate the region of each object, which is actually a bounding box and supposed to contain only one target object. Then for each object, we apply the task-oriented grasp detection in its region. However, in object stacking scenes, considering the overlaps and occlusions, it is inevitable that some non-target objects appear in the region of target object, which will greatly interfere with the results of grasp detection.

Fortunately, in task-oriented grasping, the detected grasps are supposed to satisfy the task constraints, which brings abundant semantic contents that can be explored and utilized. Firstly, for a single object, some task labels are incompatible. For example, if a grasp labelled ”screw” and a grasp labelled ”hammer” appear in an object region simultaneously, it is very likely that one of them belongs to a non-target object, because a single object is unlikely to have these two functions at the same time. And secondly, the task regions of an object are often contiguous. If a grasp labelled ”screw” appears among a cluster of grasps labelled ”hammer”, it is very likely that this grasp is a false detection. In this paper, we construct a fully connected CRF to model these semantic

Chenjie Yang 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. cjyang2017@outlook.com, xglan@mail.xjtu.edu.cn

978-1-7281-4004-9/19/$31.00 ©2019 IEEE

contents and infer it using mean field approximation [14]. More significantly, the inference process of the constructed CRF is embedded into the Deep Neural Network (DNN) as a Recurrent Neural Network (RNN), so that the whole network can be trained end-to-end. Finally, the experimental results suggest that our approach can greatly deal with the problem of task-oriented grasping in object stacking scenes.

The main contributions of this paper can be summarized as following three parts:

- A synthetic dataset OSGD is constructed for task-oriented grasping in object stacking scenes.
- A CRF is constructed to model the semantic contents in object regions, which greatly reduces the interference of overlaps and occlusions in object stacking scenes.
- The inference process of our CRF-based semantic model is embedded into the grasp detection network as a RNN, so that the whole model TOG-CRFs can be trained end-to-end and achieve a great performance.

The rest of this paper is organized as follows: The related works and problem formulation are introduced in Section II and III. The proposed approach is presented in detail in Section IV. Finally, our experiments are illustrated in Section V and the conclusions are discussed in Section VI.

II. RELATED WORK

Grasping Datasets: The traditional grasping algorithms often precompute grasps on full 3D models [15]–[17]. Based on that, they can match point clouds with these 3D models to get grasps in real world. This gave birth to some grasping datasets which annotate grasps on 3D models [18], [19]. With the development of deep learning, most recent works directly used visual data as their inputs. For this purpose, some image-based grasping datasets were collected and annotated manually, such as Cornell Grasping Dataset [20], which greatly promote the development of robotic grasping. However, as the difficulty of grasping tasks increases, the required amount of training data is growing as well [21]. To save the data collection time, many of the existing works trained their models on synthetic datasets, especially task-oriented grasping [5]–[8]. Notably, considering the difficulty of rendering photo-realistic RGB images, almost all of these works used depth maps or point clouds as their inputs.

Task-oriented Grasping: Task-oriented grasping is a widely studied topic. When performing a task, the chosen grasp pose is supposed to satisfy the task constraints, which is more challenging than just grasping with stability. Some traditional algorithms dealt with this problem by robot mechanics [1]–[4]. And in recent years, many learning-based approaches were proposed for task-oriented grasping [5]–[8], [22]–[24]. Typically, in [6], Detry et al. identified the task-compatible regions using CNN followed by the region-based detection of stable grasps. And in [7], Fang et al. trained its task-oriented grasping model by self-supervision. However, limited by the capability of policy learning, this algorithm only took two tasks into consideration. Almost all of these works are proposed for detection on single object scenes. Although [6] can deal with the situation of multiple objects in the scene, if the objects are highly stacked, the detected task-compatible regions of different objects may be connected, which will greatly interfere with its subsequent grasp detection.

Conditional Random Fields: A Conditional Random Field (CRF) is essentially an undirected Probabilistic Graphical Model (PGM). This model is widely used in computer vision, such as image semantic segmentation [25]–[28]. Although CRFs do well in modeling semantic contents, the huge time cost of inference limited their application for a long time, especially fully connected CRFs. In [25], P. Krähenbühl et al. proposed an efficient inference algorithm based on mean field approximation, which greatly reduces the inference time when the pairwise potential is the combination of gaussian kernels. And with the development of deep learning, a lot of works embedded the inference process of CRFs into neural networks [26]–[28]. In this paper, we also construct CRFs in this form, which greatly improves the performance of task-oriented grasping in object stacking scenes.

III. PROBLEM FORMULATION

The objective of this paper is to perform task-oriented grasping in object stacking scenes, which means that when given an object, for each grasp candidate of it we should regress its grasp rectangle accurately and predict its scores of stability and task classification. These three parts will be introduced in detail as follows.

Grasp Rectangle Regression: In [29], Lenz et al. proposed to use a five-dimensional vector to represent the grasp rectangle. The vector is defined as follows:

$$\mathbf{g} = (x, y, h, w, \theta) \tag{1}$$

in which (x, y) represents the location of the grasp rectangle, h is the rectangle's height which represents the gripper's open distance, w is the rectangle's width and θ denotes the angle between the rectangle's direction and the horizontal direction. This grasp representation is widely adopted by existing works on grasp detection. And in this paper, for each detected grasp, the proposed approach is supposed to regress the grasp rectangle accurately in the form of (x, y, h, w, θ).

Grasp Stability Detection: Before considering the task constraints, we should determine whether a grasp candidate is stable and belongs to a specified object. We let $\mathcal{O}$, $\mathcal{G}$, $\mathcal{C}$ represent the observation space, all grasp candidates and all objects in a scene respectively. Afterwards, we define $x_s \in \{0, 1\}$ as a binary-valued grasp stability metric, where $x_s = 1$ indicates that the target object is grasped stably. When given the observation $\mathbf{o} \in \mathcal{O}$, grasp candidate $\mathbf{g} \in \mathcal{G}$ and target object $\mathbf{c} \in \mathcal{C}$, our algorithm is supposed to model the conditional probability distribution $\mathrm{P}(x_s|\mathbf{o}, \mathbf{g}, \mathbf{c})$ which represents the probability that object $\mathbf{c}$ can be held stably by grasp $\mathbf{g}$ in view of the observation $\mathbf{o}$.

Grasp Task Classification: In this section, we let $\mathcal{T}$ represent the task space and define $x_t \in \mathcal{T}$ as a task-compatible metric, where $x_t = T$ indicates that the grasp satisfies the constraints of task T. For task-oriented grasping, the proposed approach is supposed to detect grasps that are stable and task-compatible, which means that the joint distribution

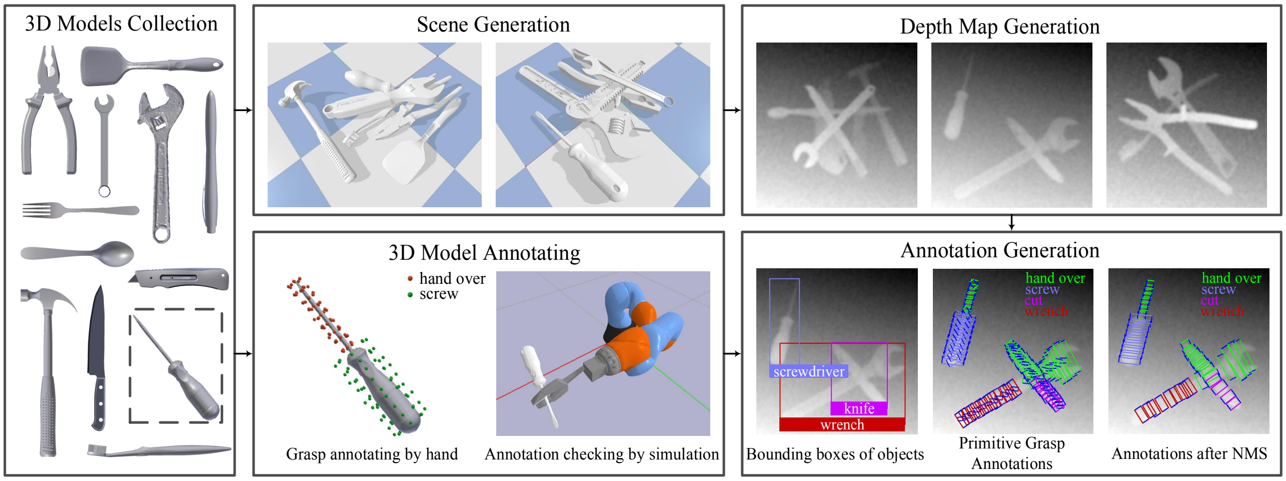


Fig. 2. Process of dataset generation. 3D Models Collection: Some 3D models that we collected. Scene Generation: Two object stacking scenes, which are produced by physics simulation in pybullet. 3D Models Annotation: The process of grasp annotation and the scene of grasp checking by simulation. Depth Map Generation: Some depth maps rendered in blender. Annotation Generation: The generated annotations in image coordinates, which include the bounding boxes of objects and the grasp rectangles assigned task labels. In this figure, grasps and task labels are connected with colors.

$\mathrm{P}(x_t, x_s|\mathbf{o}, \mathbf{g}, \mathbf{c})$ should be modelled. Because when a grasp is not stable, detecting its task label is meaningless, in this paper we only model the distribution $\mathrm{P}(x_t, x_s = 1|\mathbf{o}, \mathbf{g}, \mathbf{c})$ and decompose it as $\mathrm{P}(x_t|x_s = 1, \mathbf{o}, \mathbf{g}, \mathbf{c}) \cdot \mathrm{P}(x_s = 1|\mathbf{o}, \mathbf{g}, \mathbf{c})$ [7]. Since $\mathrm{P}(x_s = 1|\mathbf{o}, \mathbf{g}, \mathbf{c})$ has been modelled in Grasp Stability Detection, we only need to model the distribution $\mathrm{P}(x_t|x_s = 1, \mathbf{o}, \mathbf{g}, \mathbf{c})$ for grasp task classification.

IV. PROPOSED APPROACH

In this section, we firstly construct a synthetic dataset OSGD for task-oriented grasping in object stacking scenes. On this basis, a task-oriented grasping algorithm TOG-CRFs is proposed to deal with the object stacking scenes. This model detects grasps in the object regions and models the regions' semantic contents using CRFs, which makes it achieve a good performance in our task.

A. Dataset Generation

In consideration of the demand for vast training data, we construct the grasping dataset OSGD[1] by synthesis to save data collection time. The dataset contains 8000 depth maps of scenes with multiple stacked objects. For each depth map the annotations contain bounding boxes, grasp rectangles of each object and the task-compatible label of each grasp. In total, there are 10 object categories and 11 task categories in dataset OSGD, of which the tasks' constraints are shown in Table I. The process of dataset generation is illustrated in Fig. 2 and will be introduced in following parts.

3D Models' Collection and Annotation: To generate the dataset, firstly we collect 100 3D models of objects belonging to the 10 categories shown in Table I. These 3D models are selected from YCB dataset [30], ShapeNet [31], ModelNet [32] and Trimble 3D Warehouse [33]. For each model, we firstly annotate grasps that meet the task constraints by hand. For accuracy, we check all these annotated grasps by simulation grasping in pybullet [34] to filter unstable grasps.

TABLE I

TASK REQUIREMENTS

Task	Grasp Requirements	Representative Object
no task	cannot finish any task	——-
cut	avoid the blade	knife
screw	avoid the shaft	screwdriver
ladle	avoid the bowl	spoon
fork	avoid the tines	fork
wrench	away from the jaws	wrench
hammer	away from the heavier end	hammer
write	not close to the tip	pen
brush	avoid the parts with bristles	toothbrush
shovel	avoid the blade	spatula
pinch	avoid the jaws	pliers
hand over	leave the handle clear	——-

Scene Generation: To generate scenes with multiple stacked objects, we simulate stacking objects in pybullet. In this process, the objects are dropped at a height of $30\mathrm{cm}$ and the final steady state will be stored as a scene. In addition, we set the number of objects to 3-6 and we limit the objects in an area of $1\mathrm{m} \times 1\mathrm{m}$ by physical constraints.

Dataset Generation: To render depth maps and generate the annotations in image coordinate system, we load the scene generated by pybullet into blender [35]. Firstly, we read the sensor data of z channel and add zero-mean gaussian noise to get depth maps closer to real ones. Afterwards, we apply camera transformation on all vertices and grasps of the objects to get the bounding boxes and grasp rectangles in image coordinate system. Above all, same as all image-based grasping datasets, the grasp rectangles are supposed to be as parallel as possible to the camera frame, otherwise the grasps will become meaningless in image coordinate system. To

[1] http://gr.xjtu.edu.cn/web/zeuslan/tog-os-dataset

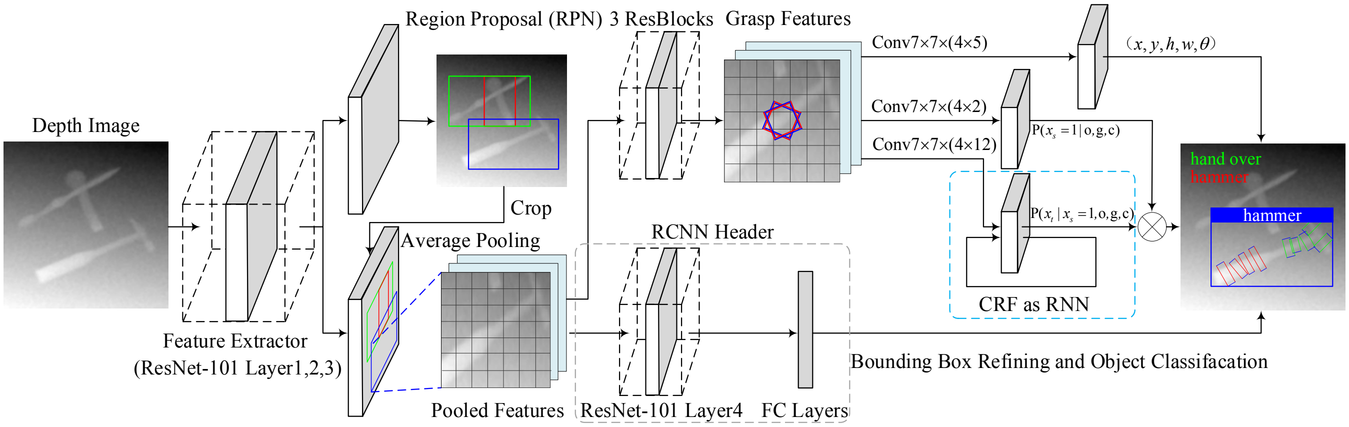


Fig. 3. Overview of TOG-CRFs. In this architecture, firstly a RPN is applied on the extracted features to generate object regions. After a 7×7 adaptive pooling, the RCNN header and the grasp detector are applied on the pooled features. The RCNN header is used to refine the bounding boxes and classify the objects. For the grasp detector, we first get the grasp features by applying 3 ResBlocks on the pooled features. Afterwards, we set 4 oriented anchors in each grid cell, which can be treated as grasp candidates. For each candidate $\mathbf{g}$ we output its regression (x, y, h, w, θ), stability score $\mathrm{P}(x_s = 1|\mathbf{o}, \mathbf{g}, \mathbf{c})$, task scores $\mathrm{P}(x_t|x_s = 1, \mathbf{o}, \mathbf{g}, \mathbf{c})$. The part in the light blue dotted box is the constructed CRF, which will be introduced in detail in Section IV-C.

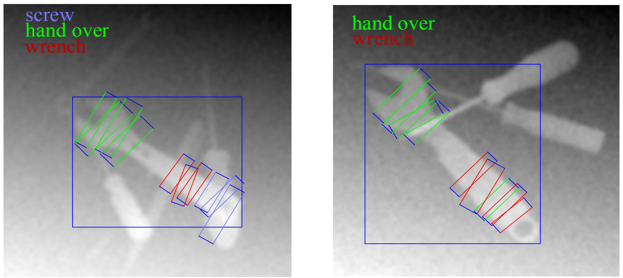
solve this problem, we firstly filter out the grasps which are less than forty-five degrees from the main optical axis of the camera. Afterwards, we apply a Non-Maximum Suppression (NMS) to the remaining grasps. This stage aims to remove the grasps that have a short Euclidean distance with other grasps which have larger angles with the main optical axis.

B. Task-oriented Grasping in Object Stacking Scenes

For our task, as illustrated in Section III, the proposed approach is supposed to provide the distributions $\mathrm{P}(x_s|\mathbf{o}, \mathbf{g}, \mathbf{c})$, $\mathrm{P}(x_t|x_s = 1, \mathbf{o}, \mathbf{g}, \mathbf{c})$ and the accurate pose regression for each grasp candidate $\mathbf{g}$. However, interfered by the overlaps and occlusions in object stacking scenes, it is difficult for the existing grasping algorithms to model them well. To solve this problem, we firstly apply task-oriented grasp detection in the object regions. Based on that, we construct a CRF to model the semantic contents of object regions, which greatly improves the performance of the proposed approach. The architecture of the constructed model is shown in Fig. 3.

The constructed model takes depth maps as input and ResNet-101 [12] as the feature extractor. On this basis, a Region Proposal Network (RPN) [13] is applied on the extracted features to get object regions. Afterwards, we use these object regions to crop the extracted features and add a 7×7 adaptive pooling layer to concatenate them as a batch. After that, there are two branches applied on the pooled features. The first one is RCNN header, which is used to refine the bounding boxes and classify the objects. The second one is the task-compatible grasp detector.

According to the size of our adaptive pooling layer, each object region can be divided into 7×7 grid cells. In each grid cell we predefine 4 oriented anchor boxes, of which each anchor box represents a grasp candidate. To expand the receptive field and reduce the scene confusion, we apply 3 ResBlocks on the pooled features of the object regions to get grasp features. After that there are 3 convolution kernels added on the grasp features. The first kernel has 4×5 output channels, which is used to regress the grasp rectangle (x, y, h, w, θ) for each grasp candidate $\mathbf{g}$. The second kernel has 4×2 output channels, which is used to model $\mathrm{P}(x_s|\mathbf{o}, \mathbf{g}, \mathbf{c})$ for each grasp candidate $\mathbf{g}$ when added a softmax layer. The third kernel has 4×12 (11 tasks plus ”no task” as shown in Table I) output channels. However, as introduced in Section I, the task labels of the grasp candidates will be greatly influenced by the semantic content. Therefore, the output of this kernel cannot be directly used to model $\mathrm{P}(x_t|x_s = 1, \mathbf{o}, \mathbf{g}, \mathbf{c})$. As an alternative, we construct a CRF to model the joint probability distribution of all grasp candidates' task labels within an object region, which will be introduced in detail in Section IV-C.



(a) Task Compatibility (b) Task Region Continuity

Fig. 4. Two kinds of task semantic contents proposed in Section IV-C. In this figure, grasps and task labels are connected with colors. (a) In this figure, the target object of the object region is a wrench while there are two detected grasps labelled ”screw”. Because ”screw” is incompatible with ”wrench” for a single object,the semantic model may be able to detect the error and suppress these two grasps. (b) In this figure, a grasp labelled ”hand over” appear among a cluster of grasps labelled ”wrench”. According to the continuity of task regions, this grasp should be suppressed.

C. CRF-based Semantic Model

In object stacking scenes, applying grasp detection in the object regions can only exclude interference outside them. However, the non-target objects within the object region will also severely interfere with the detection results. Fortunately, there are abundant semantic contents which can be utilized to further suppress the interference. This has been introduced in Section I and illustrated in Fig. 4. In this section, we will model these semantic contents using fully-connected CRFs.

According to Section III, we are supposed to model $\mathrm{P}(x_t|x_s = 1, \mathbf{o}, \mathbf{g}, \mathbf{c})$ for grasp task classification. Because in an object region, the observation $\mathbf{o}$ and the target object $\mathbf{c}$ are deterministic, we can simplify x_t to x and then the notion $\mathrm{P}(x_t|x_s = 1, \mathbf{o}, \mathbf{g}, \mathbf{c})$ to $\mathrm{P}_{\mathrm{g}}(x)$, which represents the task label's conditional probability distribution of grasp candidate $\mathbf{g}$. However, considering the existence of semantic relationships, the $\mathrm{P}_{\mathrm{g}}(x)$ of each grasp candidate $\mathbf{g}$ can't be treated as mutually independent. Therefore, instead of modelling $\mathrm{P}_{\mathrm{g}}(x)$ for each grasp candidate, we have to model probability distribution of the task labels for all grasp candidates in an object region, which we note as $\mathrm{P}(\mathbf{x})$ in this paper. $\mathbf{x}$ represents all target variables x in an object region.

When the observation and other conditions are deterministic, CRFs can be considered as a model for approximating the probability distribution $\mathrm{P}(\mathbf{x})$. Because the constructed CRFs are fully connected, all variables can be treated as one clique $\mathcal{C}$, of which the basic form can be written as:

$$\mathrm{P}(\mathbf{x}) = \frac{1}{\mathbf{Z}} \exp\left(-\sum_{i} \psi_u(x_i) - \sum_{i \neq j} \psi_p(x_i, x_j)\right) \qquad (2)$$

where x_i, x_j represent the task labels of grasp candidates $\mathbf{g}_i, \mathbf{g}_j$, $\psi_u(x_i)$ is the unary potential and used to measure the cost of assigning task label x_i to grasp $\mathbf{g}_i$, $\psi_p(x_i, x_j)$ is the pairwise potential and used to measure the cost of assigning task label x_i, x_j to grasps $\mathbf{g}_i, \mathbf{g}_j$ simultaneously and $\mathbf{Z}$ is the partition function.

As illustrated in Section IV-B, we have added a convolution kernel with 4×12 output channels on the grasp features. And we use the negative of its outputs as the unary potential ψ_u, which is an effective way to combine deep learning and CRFs [26], [27]. As for the pairwise potential ψ_p, according to the semantic context within an object region, we set:

$$\psi_p(x_i, x_j) = \omega(x_i, x_j) + \mu(x_i, x_j) k_G(\mathbf{p}_i, \mathbf{p}_j) \qquad (3)$$

where $\omega(x_i, x_j)$, $\mu(x_i, x_j)$ are learnable parameters, k_G is a gaussian kernel and $\mathbf{p}_i, \mathbf{p}_j$ represent the locations of grasp candidates $\mathbf{g}_i, \mathbf{g}_j$. The first part $\omega(x_i, x_j)$ is designed for the constraint of task compatibility. As illustrated in Fig. 4(a), an object is unlikely to have functions "screw" and "hammer" simultaneously, which can be formulized as when x_i, x_j are assigned "screw" and "hammer", $\omega(x_i, x_j)$ is supposed to be larger. The second part $\mu(x_i, x_j) k_G(\mathbf{p}_i, \mathbf{p}_j)$ is designed for the continuity of task regions and can be interpreted as the closer grasps are more likely to be assigned the same task labels, which is illustrated in Fig. 4(b). This form is firstly used for image segmentation [25] while $\mu(x_i, x_j)$ is potts model, $\mu(x_i, x_j) = 1[x_i \neq x_j]$. In this paper, we set $\mu(x_i, x_j)$ as a learnable parameter to improve the model's expression ability.

Fully connected CRFs only can be exactly inferred by traversing all values, which is time-consuming. For our problem, since each grasp has 12 possible task labels, if there are n grasp candidates in an object region, the time complexity of traversal will be $O(12^n)$, which is unacceptable. In view of this, we use mean field approximation to infer the model [14], [25]. This algorithm uses a simpler distribution $Q(\mathbf{x}) = \prod_i Q_i(x_i)$ to approximate $\mathrm{P}(\mathbf{x})$, which allows us to maximize $\mathrm{P}(\mathbf{x})$ by maximizing each approximative marginal distribution $Q_i(x_i)$. By minimizing the KL-divergence $D(Q(\mathbf{x}) \parallel \mathrm{P}(\mathbf{x}))$, we will get the update formula:

$$\begin{aligned} Q_i(x_i) = & \frac{1}{Z_i} \exp\Bigg(-\psi_u(x_i) - \sum_{x_j \in \mathcal{T}} \omega(x_i, x_j) \sum_{j \neq i} Q_j(x_j) \\ & - \sum_{x_j \in \mathcal{T}} \mu(x_i, x_j) \sum_{j \neq i} Q_j(x_j) k_G(\mathbf{p}_i, \mathbf{p}_j) \Bigg) \end{aligned} \qquad (4)$$

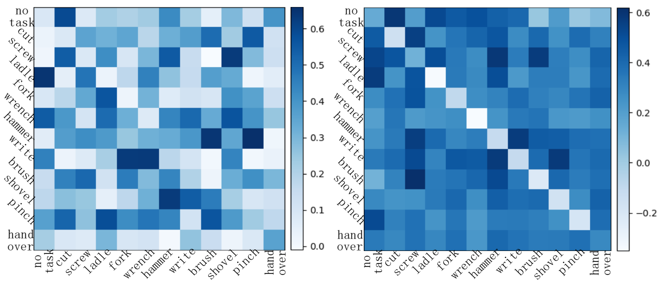
This algorithm reduces the the time complexity of inference from $O(12^n)$ to $O(n^2)$. And more importantly, the summation in this formula can be implemented as matrix multiplication, so that the update formula can be implemented as a RNN.

V. EXPERIMENTS

In this paper, we train each model 10 epochs on dataset OSGD in total and apply it on robot Baxter to perform task-oriented grasping. When training, we set the learning rate as 0.001 and the momentum as 0.9. After that, we evaluate their performance with respect to three aspects. Firstly, we visualized some parameters of our model, from which we can demonstrate the validity of the constructed CRFs. Secondly, we test our models on dataset OSGD and real depth maps, from which we can quantify the performance of our algorithm and demonstrate its generalization capability to real scenes. Above all, we test the success rate of task-oriented grasping in object stacking scenes.

A. Parameter Analysis

As illustrated in Section IV-C, our CRFs are constructed on the semantic contents of object regions. Furthermore, we embed the CRFs into the network as a RNN so that the whole model can be trained end to end. Since the training



(a) Trained $\omega(x_i, x_j)$ (b) Trained $\mu(x_i, x_j)$

Fig. 5. Visualization of the trained parameters $\omega(x_i, x_j)$ and $\mu(x_i, x_j)$. For these two heat maps, the horizontal and vertical axes all represent the task labels which are ranked as in Table I. In these heat maps, the darker colors represent the higher values.

TABLE II
ACCURACY ON TEST SET OF OUR CONSTRUCTED MODEL

Author	Algorithm	mAP	mAP-G	Acc-T (top1)	Acc-T (top5)	Acc-T (top10)	Speed (fps)
Zhou et al.	**FCGN [36] + Task**	89.2	72.9	77.9	67.8	56.9	**10.3**
Ours	**TOG**	89.2	**85.7**	93.7	83.7	76.0	9.1
	TOG-CRFsω	89.6	85.2	93.9	85.2	78.5	8.9
	TOG-CRFsμ	89.8	84.8	94.1	86.0	79.3	8.6
	TOG-CRFs	**89.9**	84.9	**95.0**	**86.1**	**79.9**	8.8

process of neural networks is uncontrollable, it is important to examine whether the trained parameters of CRFs model the semantic contents we proposed. To this end, we visualize the parameters $\omega(x_i, x_j)$, $\mu(x_i, x_j)$, as shown in Fig. 5.

For $\omega(x_i, x_j)$, a higher value means a higher penalty for assigning labels x_i, x_j to two grasps in an object region simultaneously. From the heat map of trained $\omega(x_i, x_j)$ we can see that values on diagonal and the edges are most relatively small, which implies the following two aspects. Firstly, grasps with the same task labels are more likely to appear together in an object region. Secondly, labels "hand over" are more easily allowed to coexist with other task labels. These are in line with our perception of semantic content, which demonstrates the validity of our model.

In Section IV-C, we use a learnable parameter $\mu(x_i, x_j)$ to replace the potts model. From its heat map we can see that values on diagonal are much smaller than others. This phenomenon is in line with the potts model but avoids the values of parameters being absolutely set to 0 or 1, which makes the learned parameter $\mu(x_i, x_j)$ more expressive and more adaptable than the original potts model.

B. Experiments on Dataset

Firstly, we test the constructed models on dataset OSGD and compared their performance with Fully Convolutional Grasp Detection Network (FCGN) [36], which is the state-of-the-art algorithm for stably grasping in single object scenes. In this section, we generalize FCGN to multiple object scenes by a simple combination with Faster-RCNN [13]. In addition, we add a task prediction layer on its feature map and the whole model will be noted as **FCGN + Task**. The performance metrics we used are illustrated as follows:

Mean Average Precision (mAP): In this paper, we use mAP to evaluate our models' performance of object detection, which is widely used by existing works.

Mean Average Precision with Grasp (mAP-G): This metric is used to evaluate the performance of object detection and stable grasp detection simultaneously. mAP-G is similar to mAP but a true positive detection is supposed to satisfy the following two constraints. Firstly, the detected bounding box has an Intersection over Union (IoU) larger than 0.5 with the ground truth and the object can be classified correctly. Secondly, the detected grasp with Top-1 stability score of the object must match at least one of the ground truth grasps with the Jaccard Index larger than 0.25 and the angle difference less than $30°$.

Accuracy of Task Classification (Acc-T): This metric is proposed for evaluating our models' performance on task classification while avoiding the interference of the grasp stability. For each Region of Interest (RoI), we firstly choose the grasps with top n stability scores. If all these grasps are assigned the right task labels, it will be seen as a positive example, otherwise a negative example.

The experimental results are shown in Table II. In this table, we let **TOG** represent the model that does not consider the semantic contents. And we let **TOG-CRFsω** represent the algorithm that only use parameter ω to construct CRFs, so does **TOG-CRFsμ**. **TOG-CRFs** is the constructed model that takes the complete semantic contents into consideration.

From the results we observe the following three points: 1) The performance of FCGN + Task is much lower than our models in all metrics. This further suggests that the algorithms proposed for single object grasping are not suitable for object stacking scenes. 2) TOG-CRFs has a performance similar to TOG in metrics mAP and mAP-G, this is because out CRFs is only constructed for task classification. But for Acc-T, the performance of TOG-CRFs is higher than TOG by 1.3%, 2.4%, 3.9% under the standards top1, top5 and top10 respectively. These results firstly demonstrate that the consideration of semantic contents can greatly improve the model's performance in grasp task classification. Secondly, we observe that the more grasps considered, the higher the improvement of the task classification performance. This is because we model the semantic contents with a holistic view.

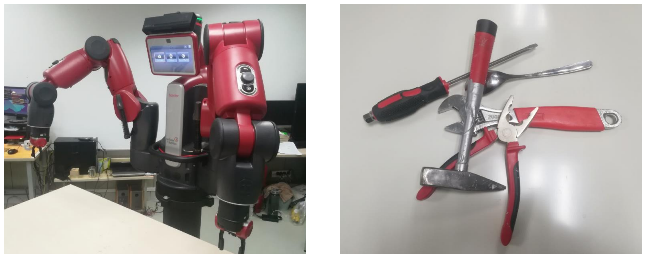

(a) Environment Configuration
(b) Object Stacking Scenes

Fig. 6. Experiment configurations. (a) A view of our environment configurations, including the Baxter robot, Kinect camera and the workbench. (b) The object stacking scenes that the robot will deal with.

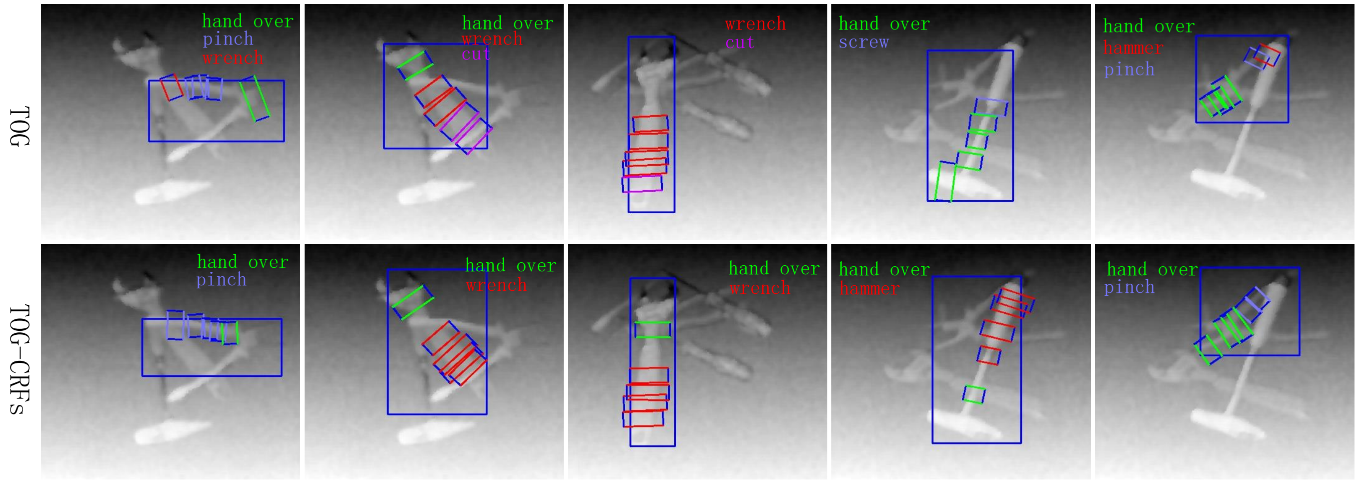


Fig. 7. Some examples of grasp detection results on real depth maps. In this figure, grasps and task labels are connected with colors. The first line is the results produced by TOG, and the second line is produced by TOG-CRFs. These detection results are generated in a way that firstly chooses the top-5 stable grasps in the detected object region and then predicts their task labels. For the first image, we can see that the grasp near the wrench is misdetected by TOG because of the overlap. And for the second image, some grasps that should be assigned "wrench" are misdetected as "cut" by TOG due to the morphological similarity. The other three images all have similar errors. But for TOG-CRFs, we can observe that all of these errors are corrected by the CRF-based semantic model, which demonstrates that the proposed approach does have a great ability to deal with the object stacking scenes in real world.

TABLE III
SUCCESS RATE OF ROBOT GRASPING

Task	**GSR-S** (%)		**GSR-T** (%)	
	TOG	**TOG-CRFs**	**TOG**	**TOG-CRFs**
no task	70	70	63.3	63.3
cut	50	50	50	50
screw	80	90	60	90
ladle	40	50	30	30
fork	40	30	20	20
wrench	80	90	60	90
hammer	100	100	80	100
write	80	80	50	50
brush	60	80	40	70
shovel	100	100	100	100
pinch	90	100	60	100
hand over	83.3	83.3	63.3	70
Total	**72.8**	**76.9**	**56.4**	**69.4**

C. Robot Experiments

We carry out our robot experiments on a Baxter which is produced by Rethink Robotics and has two 7-DoF arms with parallel jaw grippers. Since there are only color cameras installed on Baxter, in our experiments, we use an external Kinect2 to get depth maps. In addition, Baxter will manipulate the objects on a flat workbench with the size of $1\text{m}\times 1\text{m}$. The environment configuration is shown in Fig. 6(a).

In experiments, there will be 3-6 objects stacked on the workbench, as shown in Fig. 6(b). When given a task T, the robot is supposed to choose a suitable object and grasp it task-compatibly. In practice, we choose the grasp with the largest score $\text{P}(x_t = T|x_s = 1, \mathbf{o}, \mathbf{g}, \mathbf{c})$ from all detected objects to achieve this goal. Notably, because a task can be performed by more than one kind of tools, we do not take the object category into consideration when grasping. But specially, for "no task" and "hand over", because almost all of the objects have these two functions, we first randomly specify an object category and then detect the grasps only for this object category. In this section, we use the success rate on stably grasping and task-oriented grasping as the performance metrics, which will be illustrated as follows:

Success Rate for Stably Grasping (GSR-S): This metric is used to evaluate the performance of stably grasping in real-world scenes. If the robot can choose a suitable object and grasp it stably, it will be treated as a successful grasp.

Success Rate for Task-oriented Grasping (GSR-T): This metric is used to evaluate the performance of task-oriented grasping in real-world scenes. If the robot can choose a suitable object and grasp it not only stably, but task compatibly, it will be treated as a successful grasp. The requirements of each task has been illustrated in Table I.

To get the grasping success rate, for "no task" and "hand over", we implement 30 experiments respectively with a randomly specified object category. For the other 10 tasks, we implement 10 experiments for each of them. All of the experimental results are recorded in Table III. From the results we can see that, firstly, the model trained on dataset OSGD can generalize to real-world scenes well and help the robot grasp task-compatibly. And secondly, the constructed CRFs can greatly improve the model's performance on robot grasping, especially the task-oriented grasping.

When conducting the robot experiments, to intuitively demonstrate the performance of our model in real-world scenes, we draw the detection results in a way that firstly chooses the top-5 stable grasps in an object region and then predicts their task labels, as shown in Fig. 7. The first line of this figure is the results produced by TOG, and the second line is produced by TOG-CRFs. From the results we can see that in object stacking scenes, the overlaps and specular reflection may influence the detection results, but the constructed CRF-based semantic model can suppress these interference to a great extent.

VI. CONCLUSIONS

In this paper, we first construct a synthetic dataset OSGD for task-oriented grasping in object stacking scenes. Afterwards, we construct a CRF-based semantic model to suppress the interference caused by object stacking and embed its inference process into the grasp detection network as a RNN, so that the whole model TOG-CRFs can be trained end-to-end. Finally, in object stacking scenes, our model achieves a success rate of 76.9% for stably grasping and a success rate of 69.4% for task-oriented grasping. The results suggest that firstly, the synthetic dataset OSGD has a high fidelity and can help robots grasp accurately in real-world scenes. Secondly, in object stacking scenes, the consideration of semantic contents can greatly improve the model's performance on task-oriented grasping.

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.2018ZX01028-101, key project of Shaanxi province No.2018ZDCXLGY0607, and NSFC No.61573268.

REFERENCES

[1] R. Haschke, J. J. Steil, I. Steuwer, and H. J. Ritter, "Task-oriented quality measures for dextrous grasping." in *CIRA*. Citeseer, 2005, pp. 689–694.

[2] J. Aleotti and S. Caselli, "Interactive teaching of task-oriented robot grasps," *Robotics and Autonomous Systems*, vol. 58, no. 5, pp. 539–550, 2010.

[3] Y. Lin and Y. Sun, "Task-based grasp quality measures for grasp synthesis," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2015, pp. 485–490.

[4] ——, "Task-oriented grasp planning based on disturbance distribution," in *Robotics Research*. Springer, 2016, pp. 577–592.

[5] M. Kokic, J. A. Stork, J. A. Haustein, and D. Kragic, "Affordance detection for task-specific grasping using deep learning," in *2017 IEEE-RAS 17th International Conference on Humanoid Robotics (Humanoids)*. IEEE, 2017, pp. 91–98.

[6] R. Detry, J. Papon, and L. Matthies, "Task-oriented grasping with semantic and geometric scene understanding," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 3266–3273.

[7] K. Fang, Y. Zhu, A. Garg, A. Kurenkov, V. Mehta, L. Fei-Fei, and S. Savarese, "Learning task-oriented grasping for tool manipulation from simulated self-supervision," *arXiv preprint arXiv:1806.09266*, 2018.

[8] R. Antonova, M. Kokic, J. A. Stork, and D. Kragic, "Global search with bernoulli alternation kernel for task-oriented grasping informed by simulation," *arXiv preprint arXiv:1810.04438*, 2018.

[9] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[10] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[11] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.

[12] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[13] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, 2015, pp. 91–99.

[14] D. Koller, N. Friedman, and F. Bach, *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

[15] A. T. Miller and P. K. Allen, "Graspit! a versatile simulator for robotic grasping," 2004.

[16] D. Prattichizzo and J. C. Trinkle, "Grasping," *Springer handbook of robotics*, pp. 671–700, 2008.

[17] F. T. Pokorny and D. Kragic, "Classical grasp quality evaluation: New algorithms and theory," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2013, pp. 3493–3500.

[18] C. Goldfeder, M. Ciocarlie, H. Dang, and P. K. Allen, "The columbia grasp database," 2008.

[19] A. Kasper, Z. Xue, and R. Dillmann, "The kit object models database: An object model database for object recognition, localization and manipulation in service robotics," *The International Journal of Robotics Research*, vol. 31, no. 8, pp. 927–934, 2012.

[20] "Cornell grasping dataset," http://pr.cs.cornell.edu/grasping/rect_data/data.php, 2013.

[21] N.-n. Zheng, Z.-y. Liu, P.-j. Ren, Y.-q. Ma, S.-t. Chen, S.-y. Yu, J.-r. Xue, B.-d. Chen, and F.-y. Wang, "Hybrid-augmented intelligence: collaboration and cognition," *Frontiers of Information Technology & Electronic Engineering*, vol. 18, no. 2, pp. 153–179, 2017.

[22] D. Song, K. Huebner, V. Kyrki, and D. Kragic, "Learning task constraints for robot grasping using graphical models," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2010, pp. 1579–1585.

[23] L. Antanas, P. Moreno, M. Neumann, R. P. de Figueiredo, K. Kersting, J. Santos-Victor, and L. De Raedt, "High-level reasoning and low-level learning for grasping: A probabilistic logic pipeline," *arXiv preprint arXiv:1411.1108*, 2014.

[24] D. Song, C. H. Ek, K. Huebner, and D. Kragic, "Task-based robot grasp planning using probabilistic inference," *IEEE transactions on robotics*, vol. 31, no. 3, pp. 546–561, 2015.

[25] P. Krähenbühl and V. Koltun, "Efficient inference in fully connected crfs with gaussian edge potentials," in *Advances in neural information processing systems*, 2011, pp. 109–117.

[26] C. Liang-Chieh, G. Papandreou, I. Kokkinos, K. Murphy, and A. Yuille, "Semantic image segmentation with deep convolutional nets and fully connected crfs," in *International Conference on Learning Representations*, 2015.

[27] S. Zheng, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang, and P. H. Torr, "Conditional random fields as recurrent neural networks," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1529–1537.

[28] M. T. Teichmann and R. Cipolla, "Convolutional crfs for semantic segmentation," *arXiv preprint arXiv:1805.04777*, 2018.

[29] I. Lenz, H. Lee, and A. Saxena, "Deep learning for detecting robotic grasps," *The International Journal of Robotics Research*, vol. 34, no. 4-5, pp. 705–724, 2015.

[30] B. Calli, A. Singh, A. Walsman, S. Srinivasa, P. Abbeel, and A. M. Dollar, "The ycb object and model set: Towards common benchmarks for manipulation research," in *2015 international conference on advanced robotics (ICAR)*. IEEE, 2015, pp. 510–517.

[31] A. X. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, *et al.*, "Shapenet: An information-rich 3d model repository," *arXiv preprint arXiv:1512.03012*, 2015.

[32] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao, "3d shapenets: A deep representation for volumetric shapes," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1912–1920.

[33] *Trimble 3D Warehouse*. [Online]. Available: https://3dwarehouse.sketchup.com/

[34] E. Coumans and Y. Bai, "pybullet, a python module for physics simulation for games, robotics and machine learning," http://pybullet.org/, 2016–2017.

[35] Blender Online Community, *Blender - a 3D modelling and rendering package*, Blender Foundation, Blender Institute, Amsterdam, 2016. [Online]. Available: http://www.blender.org

[36] X. Zhou, X. Lan, H. Zhang, Z. Tian, Y. Zhang, and N. Zheng, "Fully convolutional grasp detection network with oriented anchor box," in *Intelligent Robots and Systems (IROS), 2018 IEEE/RSJ International Conference on*. IEEE, 2018.