A Learning Framework for Robust Bin Picking by Customized Grippers

Yongxiang Fan, Hsien-Chung Lin, Te Tang, Masayoshi Tomizuka

***Abstract*— Customized grippers have specifically designed fingers to increase the contact area with the workpieces and improve the grasp robustness. However, grasp planning for customized grippers is challenging due to the object variations, surface contacts and structural constraints of the grippers. In this paper, we propose a learning framework to plan robust grasps for customized grippers in real-time. The learning framework contains a low-level optimization-based planner to search for optimal grasps locally under object shape variations, and a high-level learning-based explorer to learn the grasp exploration based on previous grasp experience. The optimization-based planner uses an iterative surface fitting (ISF) to simultaneously search for optimal gripper transformation and finger displacement by minimizing the surface fitting error. The high-level learning-based explorer trains a region-based convolutional neural network (R-CNN) to propose good optimization regions, which avoids ISF getting stuck in bad local optima and improves the collision avoidance performance. The proposed learning framework with RCNN-ISF is able to consider the structural constraints of the gripper, learn grasp exploration strategy from previous experience, and plan optimal grasps in clutter environment in real-time. The effectiveness of the algorithm is verified by experiments.**

I. Introduction

Customized grippers have been broadly applied in industry to execute complex tasks such as assembly and packaging. Compared with general parallel grippers, customized grippers usually consist of curved fingers and have large contact surfaces to match with the geometries of the objects. Therefore, the customized grippers can generate more stable and robust grasps compared with those generated by general parallel grippers.

However, the grasp planning for customized grippers is challenging. On one hand, it has potentially large contact areas with workpieces, thus the traditional point contact model assumption [1] and the quality metrics [2], [3] built upon the point contact model are not applicable. As a result, the performance of the grasp planning algorithms based on those metrics [4], [5] would be downgraded. On the other hand, the grasp policies learned from end-to-end manner [6], [7] use general parallel grippers with millions of grasping data, and changing fingertips require re-collecting data and re-training the policies. With the tend of mass customization and requirement to change fingertips frequently, the end-to-end learning becomes less efficient.

There are a few works that consider the surface information during grasp planning. In [8], the optimal grasp is searched by enclosing the object with more contacts to match the object surface. However, it has a heavy computation load, thus is not suitable for online implementation. The grasp synthesis of human hands with shape matching is introduced in [9]. However, this approach requires offline human demonstration and exhaustive search in the database. In [10], [11], the curve of the parallel gripper is matched to the surface of objects to speed up the searching process. However, it cannot be applied to the precise matching for customized grippers with complicated shapes.

In this paper, we propose a learning framework to plan grasps for customized grippers. To consider more complicated gripper shapes and retrieve reliable and secure grasps for different objects, it is desired to match the surfaces on the gripper to the object more precisely. We use an iterative surface fitting (ISF) algorithm proposed in our previous work [12] to fit the contact surface of multiple fingers to the object surfaces while satisfying the structural constraints. A guided sampling is also introduced to avoid the local optima and encourage the exploration of the regions with high fitting scores. ISF achieves efficient and precise grasp planning for a single object or slight clutter environment with small searching regions. However, the searching becomes less efficient in heavy clutter environment due to the excessively sampling candidates [12].

To improve the efficiency of grasp planning in a heavy clutter environment, we propose a topological learning approach using regions with convolutional neural networks (R-CNN) [13] to learn the essential features that affect the successful execution of grasps. The features include the spatial relationships between objects which are generally difficult to model. Thus the input to the CNN is the patches of the candidate regions. This learning-based planner is connected hierarchically with ISF in order to reduce the effect of object variations and plan reliable/precise grasps under data shortage. Therefore, the learning framework includes an optimization-based planner, which searches for precise grasp pose based on the fine details of the selected region using ISF, and a learning-based explorer, which learns the desired regions to start ISF searching.

Compared with end-to-end learning methods [4], [6], the proposed learning-based explorer ignores the details of grasp planning by detecting the desired regions within the image plane for potentially high fitting scores and better collision avoidance performance, thus the dimension of the learning module is lower than end-to-end learning. The optimization-based planner searches for optimal grasps precisely in the chosen region based on the object-specific features, which are generally not shared across objects and difficult to learn by end-to-end manner. Therefore, the proposed learning

Yongxiang Fan and Masayoshi Tomizuka are with University of California, Berkeley yongxiang_fan, tomizuka@berkeley.edu
Hsien-Chung Lin and Te Tang are with FANUC Advanced Research Lab.
This work was supported by FANUC Corporation.

framework is able to improve the learning efficiency and performance at the same time.

The contributions of this paper are as follows. First, the proposed low-level optimization-based planner includes both the fingertip surfaces and the structural constraints of the gripper such as the jaw width and the allowable degree-of-freedoms (DOFs). It also achieves simultaneous surface fitting and gripper kinematic planning. Second, by combining with the dedicated gripper design, the proposed surface fitting algorithm can deal with objects with complicated shapes as well as a clutter task with unsegmented point clouds. Furthermore, the proposed grasp exploration strategy avoids getting stuck in the local optima by learning from the previous grasping experience. The grasp planning by ISF and R-CNN achieves a real-time planning and the time to search for a collision-free grasp is less than 0.3 s for the objects in clutter environment. The experimental videos are available at [14].

The remainder of this paper is described as follows. The grasp planning problem for customized gripper is stated in Section II, followed by the introduction of the proposed learning framework in Section III. The experimental verification of the proposed framework on grasp planning with customized grippers are presented in Section IV. Section V concludes the paper and proposes future works. The experimental videos are available at [14].

II. Problem Statement

The objective of the grasp planning is to search for the optimal gripper pose and finger configuration by maximizing a quality metric considering the constraints of the customized grippers. A grasp planning example is illustrated in Fig. 1. The customized gripper has parallel jaws with customized curved fingertip surfaces. Taking the gripper in Fig. 1 as an example, the grasp planning problem is defined as

$$\max_{R,t,\delta d,\mathcal{S}_j^f,\mathcal{S}_j^o} \quad Q(\mathcal{S}_1^f,\mathcal{S}_2^f,\mathcal{S}_1^o,\mathcal{S}_2^o) \tag{1a}$$

$$s.t. \quad \mathcal{S}_j^f \subset \mathcal{T}(\partial\mathcal{F}_j;R,t,\delta d), \quad j=1,2 \tag{1b}$$

$$\mathcal{S}_j^o = NN_{\partial\mathcal{O}}(\mathcal{S}_j^f), \quad j=1,2 \tag{1c}$$

$$(\mathcal{S}_1^f,\mathcal{S}_2^f) \in \mathcal{W}(d_0+\delta d) \tag{1d}$$

$$d_0+\delta d \in [d_{\min},d_{\max}] \tag{1e}$$

where $j \in \{1,2\}$ is the finger index, $R \in SO(3), t \in \mathbb{R}^3$ are the rotation and translation of the gripper jaw, $\delta d \in \mathbb{R}$ is the jaw displacement from the original width d_0, and Q is the quality metric in terms of the finger contact surfaces $\mathcal{S}_j^f$ and the object contact surface $\mathcal{S}_j^o$. The finger contact surface $\mathcal{S}_j^f$ lies on the finger surface $\partial\mathcal{F}_j$ transformed by $\mathcal{T}$ with amount of R,t, as shown in (1b). The object contact surface $\mathcal{S}_j^o$ is defined by searching the nearest neighbor of the $\mathcal{S}_j^f$ on the object surface $\partial\mathcal{O}$, as shown in (1c). Constraint (1d) indicates that the finger contact surfaces should be constrained in the space $\mathcal{W}$ parameterized by the jaw width, and (1e) describes the finger displacement constraint.

Problem (1) would be a standard grasp planning if the contact surface degenerates into a single contact point.

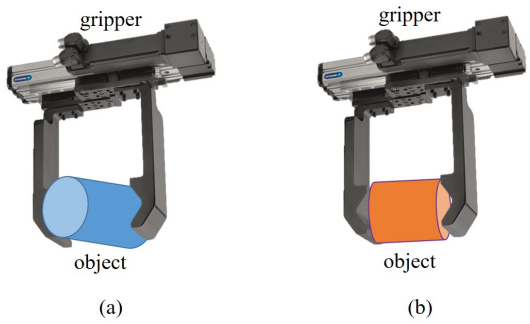


Fig. 1. An grasp example by a customized gripper with curved fingertip surfaces. A natural quality metric is the surface fitting error between the gripper and the object, where the blue object in (a) has more steady grasp compared with the orange object in (b).

In general, however, the point contact model may not be able to directly include gripper surface into the planning. Problem (1) is challenging to solve by either learning or optimization. On one hand, the learning requires training on objects with large variety. On the other hand, the optimization with gradient or sampling can either be nontrivial to use contact surfaces as decision variables, or require to search the whole state space, which is not appropriate for real-time implementation.

A natural surface-related quality can be constructed by matching the surfaces of the gripper towards the object, as shown in Fig. 1. Intuitively, the grasp with small surface fitting error (Fig. 1(a)) is more reliable and robust compared with the one with large surface fitting error (Fig. 1(b)). With the surface fitting error as the quality metric, the problem can be addressed by point set registration algorithm, such as the rigid registration (e.g. ICP [15]) or non-rigid registration (e.g. CPD [16] or TPS [17]). The non-rigid registration allows arbitrary deformation of the point set without considering the structural constraints (Constraint 1d 1e), while the rigid registration assumes rigid transformation of the point set without considering the allowable motion between different portions of the points. Both the methods tend to be trapped into local optima during the searching.

III. The Learning Framework

A. Overview

In this paper, we assume that the customized gripper has rigid fingertip surfaces and allows motion in certain DOFs. Therefore, the proposed searching algorithm is modified from the rigid registration by including the structural constraints of different fingers such as the width and DOFs. To avoid being trapped into local optima and achieve better collision avoidance performance, we propose a learning-based explorer to detect the desired regions to initialize the low-level search.

Figure 2 shows the overall learning framework. The proposed learning framework decouples the end-to-end learning into a low-level optimization-based planner and a high-level learning-based explorer. The optimization-based planner searches for optimal gripper pose and finger configuration

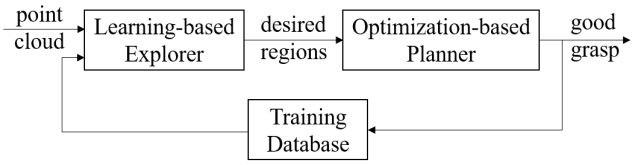


Fig. 2. Block diagram of the overall learning framework with RCNN.

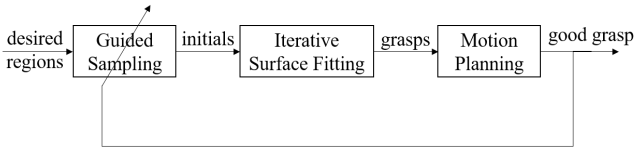


Fig. 3. Block diagram of the low-level optimization-based planner.

with the iterative surface fitting (ISF) we proposed in [12]. The learning-based explorer detects the desired region to explore from the previous grasping experiences with the regions with R-CNN. Compared with end-to-end learning methods, the optimization-based planner is more precise and reliable when grasping on various unknown objects, and the learning-based explorer is more efficient with much less training data and lower learning dimension.

B. Optimization-based Planner

The optimization-based planner searches for the optimal gripper pose and finger configuration within some particular regions on the object surface. The block diagram of the optimization-based planner is shown in Fig. 3. It contains an iterative surface fitting (ISF) and a guided sampling. ISF is to register multiple fingertip surfaces to the target workpiece considering the allowable motions of fingers, and the guided sampling is to guide the ISF to search in different portions within the region to reduce the effect of the local optima.

1) Iterative Surface Fitting: The algorithm is illustrated by a customized gripper with two fingers and one DOF, where the two fingers move in opposite directions to adjust the jaw width, as shown in Fig. 4 (Left). ISF iteratively executes two modules: the correspondence matching and the surface fitting. The correspondence matching contains the nearest neighbor search and outlier/duplicate filtering, as shown in Fig. 4 (Right)(a). The surface fitting searches over the gripper transformation (R, t) and the jaw width

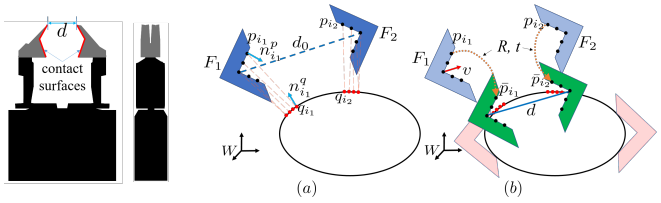


Fig. 4. (Left) Customized gripper used in the paper, and (Right) illustration of the iterative surface fitting (ISF). (a) The correspondence matching, where the corresponding point q_{i_j} on the object surface is found by searching with the nearest neighbor and removing outliers and duplicates. (b) The surface fitting, in which the gripper transformation (R, t) and finger displacement δd are optimized. The green gripper is updated by the optimization result, and the pink one is the converged result after several iterations of ISF.

Algorithm 1 Iterative Surface Fitting (ISF)

1: **Input:** Initial state R_c, t_c, d_0, $\partial\mathcal{O}$, $\partial\mathcal{F}$, L, I_0, ϵ_0
2: **Init:** $\partial\mathcal{F} = \mathcal{T}(\partial\mathcal{F}; R_c, t_c, d_0)$
3: **for** $l = L-1, \cdots, 0$ **do**
4: $\quad\mathcal{S}^f \leftarrow \texttt{downsample}(\partial\mathcal{F}, 2^l)$, $I_l = I_0/2^l$, $\epsilon_l = 2^l\epsilon_0$
5: $\quad\mathcal{S}_0^f \leftarrow \mathcal{S}^f$, $e_s \leftarrow \infty$, $\eta \leftarrow 0$
6: $\quad$**while** $\eta \notin [1-\epsilon_l, 1+\epsilon_l]$ and $it{+}{+} \le I_l$ **do**
7: $\quad\quad e_{s,p} \leftarrow e_s$
8: $\quad\quad\mathcal{S}^o \leftarrow NN_{\partial O}(\mathcal{S}^f)$
9: $\quad\quad\{\mathcal{S}^f, \mathcal{S}^o\} \leftarrow \texttt{filter}(\mathcal{S}^f, \mathcal{S}^o)$
10: $\quad\quad\{R^*, t^*, \delta d^*, error\} \leftarrow \textbf{IPFO}(\mathcal{S}^f, \mathcal{S}^o, d_0)$
11: $\quad\quad\mathcal{S}^f \leftarrow \mathcal{T}(\mathcal{S}^f; R^*, t^*, \delta d^*)$
12: $\quad\quad\partial\mathcal{F} \leftarrow \mathcal{T}(\partial\mathcal{F}; R^*, t^*, \delta d^*)$
13: $\quad\quad d_0 \leftarrow d_0 + \delta d^*$
14: $\quad\quad e_s \leftarrow \|\mathcal{S}^f - \mathcal{S}_0^f\|$, $\eta \leftarrow e_s/e_{s,p}$
15: $\quad$**end while**
16: **end for**
17: **return** $\{error, \partial\mathcal{F}\}$

$d = d_0 + \delta d$, as shown in Fig. 4 (Right)(b). More specifically, the surface fitting is to minimize the surface fitting error E:

$$\min_{R,t,\delta d} \; E(R, t, \delta d) \tag{2a}$$

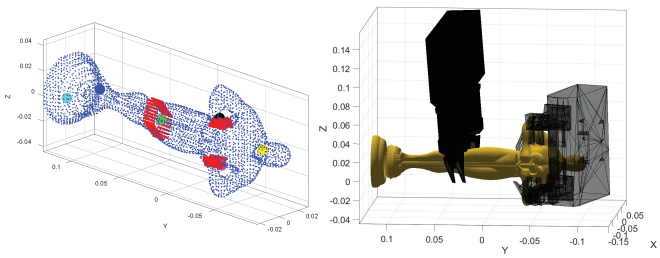
$$s.t. \quad \delta d + d_0 \in [d_{\min}, d_{\max}] \tag{2b}$$

Problem (2) is nonlinear due to the coupling between the R and δd and can be solved by an iterative palm-finger optimization (IPFO) [12].

The ISF algorithm is summarized in Alg. 1. Inspired by [18], ISF is optimized hierarchically by searching with a multi-resolution pyramid. The inputs to ISF include the initial gripper state R_c, t_c, d_0, the surfaces $\partial\mathcal{O}$ and $\partial\mathcal{F}$. L, I_0 and ϵ_0 denote the level number, maximum iteration and error bound for convergence, respectively. The gripper surface ∂F is first transformed to the specified initial state (Line 2). In each level of pyramid, the $\partial\mathcal{F}$ is downsampled adaptively to $\mathcal{S}^f$ with different resolutions (Line 4). The while loop iteratively searches correspondence and solves for desired palm transformation and finger displacement. The while loop is terminated if IPFO gives a similar transformation in adjacent iterations.

2) Baseline Initialization and Guided Sampling: The initialization of ISF is important since ISF converges to local optima. Various initial points are desired for ISF to explore different portions of the region, so as to avoid getting trapped in bad local optima and achieve better collision-avoiding solutions. In the baseline initialization, the region is partitioned into K clusters by k-means clustering. The center of each cluster is regarded as a candidate initial position of the gripper for ISF. The ISF with this baseline initialization is called baseline-ISF in the remaining of the paper.

We build an empirical model to guide the sampling among K candidates. Similar to the multi-armed bandit model, we record the fitting error, collision of ISF, and the reachability for each cluster and compute the average regret accordingly. The weight for ISF to be initialized in the k-th center is



(a) (b)

Fig. 5. (a) Illustration of the grasp planning on an Oscar model, where the blue and red dots represent the object and the gripper surfaces, and the bold dots represent the centers of k-means. (b) The visualization of planned grasps, where the collided grasp is represented by the transparent one.

decreased if this center has larger regret value, and the cluster center with the minimum regret is chosen as the initial position of the gripper for the following ISF, while the initial orientation R_c is randomly sampled.

Figure 5 shows the baseline-ISF result on an Oscar model. The object and gripper surfaces are fed into the algorithm as shown by the blue and red dots in Fig. 5(a), after which the k-means clustering ran for centers of initialization, as shown by bold dots. Multiple grasps were generated (red patches in Fig. 5(a)) and passed through the collision check function. The planned collided grasp is shown with transparency, as shown in Fig. 5(b). The centers with small surface fitting error and no collision would have small regret values, thus will be sampled more frequently.

The optimization-based planner (baseline-ISF) with the k-means clustering and guided sampling is able to search the optimal collision-free grasps efficiently in simple environment with small regions. However, the searching would be less efficient and sub-optimal in clutter environment where an excessively large number of clusters are required to be searched. Moreover, while the guided sampling distinguishes different centers of clusters by taking into account of the fitting error, collision and reachability, this allocation only explores around the predefined K positions without considering other portions of the region. Finally, the guided sampling can only exploit the experience of the current grasp searching in the current environment, since the previous grasping is conducted in different environment with different distribution of centers and regrets.

C. Learning-based Explorer

The optimization-based planner solved by baseline-ISF is inefficient and sub-optimal in clutter environment. Meanwhile, we found that human tends to decouple the process of choosing the desired grasp region from that of searching specific grasp poses inside that region to increase the efficiency of the grasping. With this observation, we design a learning-based explorer to initialize the baseline-ISF search. The explorer selects the regions with potential low regret based on the previous grasp experiences. We use R-CNN to learn a classifier in order to detect the desired regions for initialization.

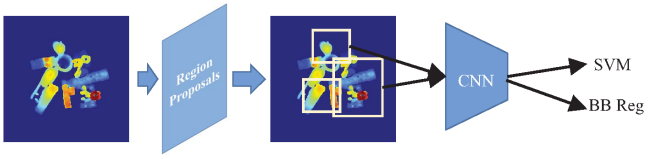
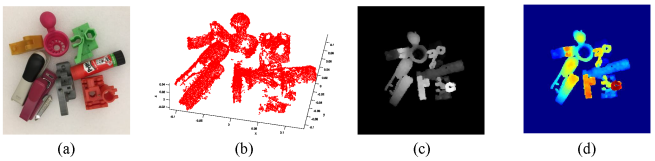


Fig. 6. Illustration of R-CNN pipeline. A region proposal block proposes regions and sends to CNN to extract features. SVMs and bounding box regression are applied to classify the proposed region and refine the bounding boxes, respectively.



(a) (b) (c) (d)

Fig. 7. The depth rendering. (a) Original scene. (b) Point cloud observed by two stereo cameras. (c) Rendered depth images. The closer of the points to the camera, the more white they are in the depth image. (d) The image is further rendered into jet colormap.

1) R-CNN Pipeline: The pipeline of R-CNN is shown in Fig. 6. R-CNN is first introduced in [13] for object detection. R-CNN contains a region proposal block to provide possible choices of regions. The region proposal selects 2K regions with different sizes using the method such as selective search. The regions are resized and fed into a CNN for feature extraction. The CNN can be pre-trained by AlexNet [19] or VGG-16/19 [20]. The outputs of CNN are used to represent the features of the region proposals. SVMs are applied to classify the regions and bounding box regression is applied to further correct the positions of the bounding boxes.

2) R-CNN Training: We use a grasping pool with 25 objects. We randomly choose some of the objects and place them in the workspace. The scene is observed by two stereo cameras and the collected point cloud is rendered as depth images, as shown in Fig. 7. R-CNN takes the rendered depth images as inputs, and produces regions of interest (ROI) to initialize the ISF searching. The ROI used for training is generated based on the optimal grasps found from baseline-ISF. The training process is illustrated by Fig. 8. The R-CNN in this paper is pre-trained by AlexNet and is fine-tuned by the data we collected. In this stage, we use 250 data pairs to fine tune the network. Some of the data pairs are illustrated in Fig. 9.

3) R-CNN Testing: At the test time, the trained R-CNN is applied to generate the ROI on the colormap. The desired regions in the point cloud are then computed based on the box coordinates of ROI in the image plane as well as their depth values. The centers of these regions are regarded as the good initialization and ISF searches from these initial positions. The proposed learning ISF is called RCNN-ISF and the framework is illustrated in Fig. 10. The proposed learning framework with RCNN-ISF not only searches grasps locally within a small region, but also learns the large-scale grasp exploration using R-CNN. Therefore, it tends to provide better initialization more efficiently than the baseline-ISF.

Compared with end-to-end learning [4], [6], the proposed

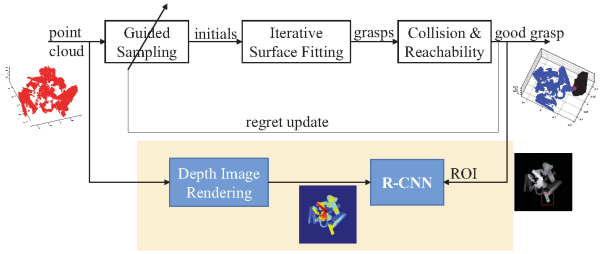


Fig. 8. Illustration of the training framework. The framework contains the baseline-ISF and the R-CNN training.

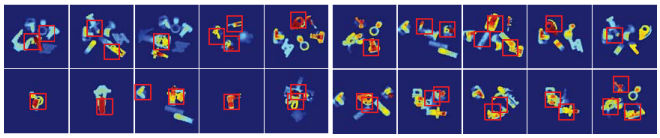


Fig. 9. The data used to fine-tune the R-CNN for good region detection.

RCNN-ISF method has the following advantages. First, the learning dimension of the RCNN-ISF is generally lower than that of the end-to-end learning. More specifically, RCNN-ISF searches ROI in the two-dimensional image plane, while the end-to-end learning searches over higher dimension depending on the grasps and grippers. For example, a grasp planning for a eight-DOF hand with three fingers has 32 dimensions [21]. Therefore, the end-to-end learning requires much more data than the proposed method. Secondly, ISF searches for optimal grasps based on object-specific features that are not shared cross objects, instead of learning the behavior end-to-end from millions of data. Consequently, ISF tends to generate more precise and robust grasps. Moreover, RCNN-ISF is able to produce versatile grasp as the experiment shows. On the contrary, the learned networks in [4], [6] produce planar grasps with simple parallel grippers.

IV. Experimental Results

This section presents the experimental results for baseline-ISF and RCNN-ISF to verify the effectiveness of the learning framework. The experimental videos are available at [14]. The host computer we used was a desktop with 32GB RAM and 4.0GHz CPU. All the computations were conducted by Matlab. We used a SMC LEHF20K2-48-R36N3D parallel jaw gripper with the specialized curved fingers as shown in Fig. 4 (Left). The desired contact surfaces are marked by red. The gripper was equipped on a FANUC LRMate 200iD/7L industrial manipulator. Two IDS Ensenso N35 stereo camera sets were used to capture the point cloud of the object. The point cloud was smoothed and the normal vectors were calculated. Despite the preprocessing, the point cloud produced by Ensenso cameras was not able to reflect the object precisely due to occlusion and noise.

A. Parameter Lists

The gripper width was constrained by $[d_{\min}, d_{\max}] = [1, 3]$ cm. The initial gripper width was set as $d_0 = 2$ cm. The pyramid level L, the maximum iteration I_0 and the tolerance ϵ_0 were set as 4, 200 and 0.008 respectively in Alg. 1. The k-means center number $K = 6$.

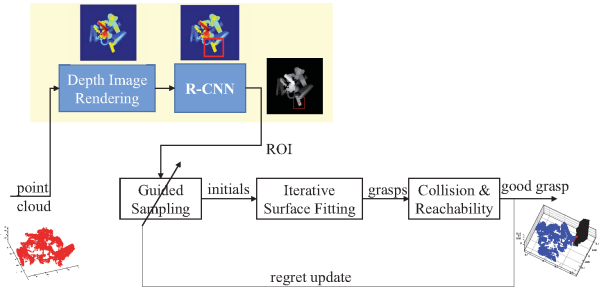


Fig. 10. The learning framework with RCNN-ISF implementation. The R-CNN detects desired low regret regions and ISF searches grasps on the selected regions.

B. Baseline-ISF Experiment

This section presents the experimental results of the optimization-based planner with baseline-ISF. The baseline-ISF algorithm considers the surface fitting around different initial positions specified by k-means. The guided sampling enables more effective exploration based on the grasp experience of the current environment.

Figure 11 shows the grasp planning results on six different objects. The left and right sides of each subfigure show the grasp optimized by ISF and the result in lifting the object by 10 cm, respectively. Baseline-ISF was able to find a grasp that matched the fingertips to the object in spite of the complicated shapes of the surface. The algorithm was able to handle the missing points properly. For example, the gripper fingertip was able to locate to the bottom of the link (Fig. 11(b)), despite the points were missing since both cameras were looking from the top. The geometry similarity between the gripper and the object is beneficial for the gripper to resist the effect of calibration uncertainty and external disturbance, and increase the stability and the payload, as it increases the potential contact surfaces between the gripper and the object. In average, it takes less than 1.5 secs to find 10 collision-free optimal grasps.

C. RCNN-ISF Experiment

This section presents the experimental results of the proposed learning framework with RCNN-ISF. The experimental videos are available at [14]. The learning framework includes the low-level optimization-based planner with baseline-ISF, and the high-level learning-based explorer with R-CNN for grasp exploration in heavy clutter environment. The training process of R-CNN is shown in Fig. 9. Same experimental setup was used as Section IV-B.

Figure. 12 illustrates the performance of the RCNN-ISF in light clutter environment. In this environment, RCNN-ISF achieved comparable performance with the baseline-ISF. The initial configuration of the object set is shown in Fig. 12(a). Figure 12(b)-(f) show the consecutive grasps in the task. The left side of each subfigure is the depth image and the R-CNN output of the confidence map for the desired regions, after which the baseline-ISF was performed on the chosen regions, and the best grasp was executed by the FANUC industrial manipulator, as shown in the right side of each subfigure. Even though the surface composed by

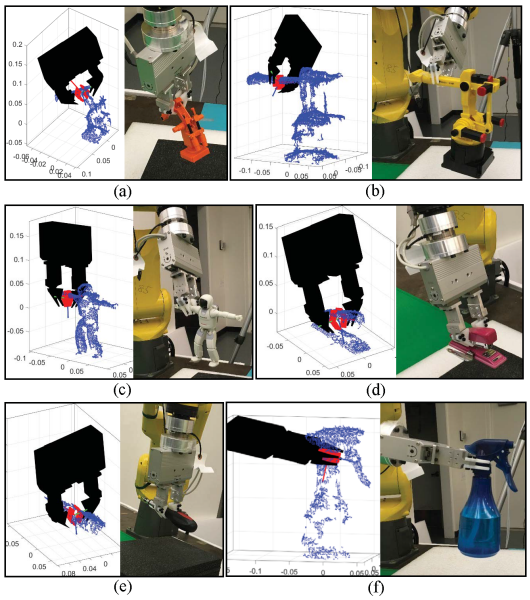


Fig. 11. The results of the grasp planning experiment on six objects.

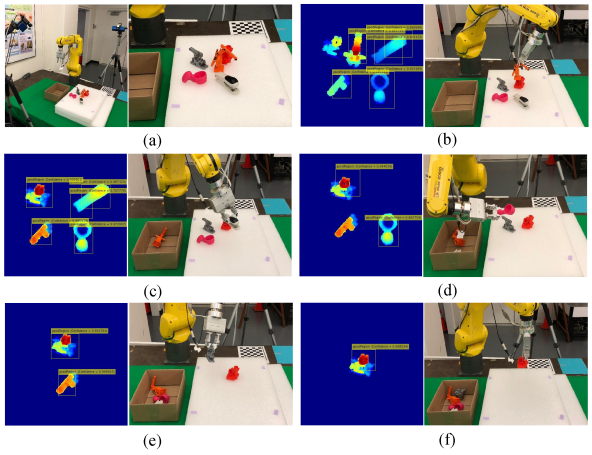


Fig. 12. Grasp planning experiment in a clutter environment. (a) The initial object clutter. (b)-(f) The consecutive grasps in the task.

the cluttered objects became more complicated than a single object, RCNN-ISF was able to successfully detect the desired regions for grasping and search on the selected regions for the optimal collision-free grasp.

Figure 13 shows the grasp planning results on heavy clutter environments by RCNN-ISF. The first row shows different clutter environments. R-CNN took rendered depth images as inputs and generated the desired regions to initialize ISF searching, as shown in Fig. 13 (Middle). The optimization-based planner produced optimal grasp pose by minimizing the fitting error and checking the collision/feasibility. The optimal grasps were executed by the FANUC manipulator, as shown in Fig. 13 (Bottom).

The comparison between the baseline-ISF and RCNN-ISF is shown in Table I for the clutter environments in Fig. 13. The searching on clutter environments are more difficult due to the collision with environments and the excessive search space for guided sampling. Consequently, the initialization becomes more important for efficient exploration. The baseline-ISF requires to initialize and execute ISF multiple times in order to explore broader regions. On the contrary, RCNN-ISF employs the previous experience for initialization and generates only a few low-regret regions to start ISF searching. Therefore, RCNN-ISF is able to perform grasp planning more efficiently. In heavy clutter environment, the baseline-ISF spent 17.23 secs to find 1 optimal grasp, while the proposed RCNN-ISF spent 1.52 secs to find 7 optimal grasps.

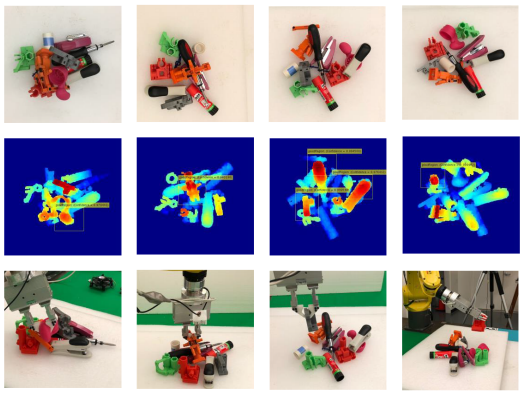


Fig. 13. Grasp planning results on clutter environment by RCNN-ISF. (Up) Clutter environment. (Middle) Selected regions and their confidence value by R-CNN to initialize ISF searching. (Bottom) Execution of the grasps planned by ISF.

TABLE I
COMPARISON OF BASELINE-ISF AND RCNN-ISF

Methods	Baseline-ISF	RCNN-ISF
Search Time (s)	17.23	1.52
Found Grasps#	1	7

V. CONCLUSIONS AND FUTURE WORKS

This paper proposed a learning framework to plan robust grasps for customized grippers. The learning framework includes a low-level optimization-based planner and a high-level learning-based explorer. The optimization-based planner used an iterative surface fitting (ISF) with guided sampling to search for optimal grasps by minimizing the surface fitting error. The performance of this low-level planner was locally effective and was sensitive to initialization. Therefore, the learning-based explorer was introduced with a region-based convolutional neural network (R-CNN) to search for desired low-regret regions to initialize ISF search. A series of experiments on robotic bin picking were performed to evaluate the proposed method. Experimental results show that the proposed learning framework with RCNN-ISF achieved a more efficient planning on heavy clutter environment, by significantly decreasing the average searching time from 17.23 secs to 1.52 secs.

There are several directions that we want to explore in the future. First, we would like to apply the algorithm to different types of grippers. In addition, we would like to extend the algorithm to the grasp planning of general multi-fingered hands and soft hands, by simultaneously searching for optimal transformation and finger joint configuration.

REFERENCES

[1] R. M. Murray, Z. Li, and S. S. Sastry, *A mathematical introduction to robotic manipulation*. CRC press, 1994.

[2] C. Ferrari and J. Canny, "Planning optimal grasps," in *Robotics and Automation, 1992. Proceedings., 1992 IEEE International Conference on*. IEEE, 1992, pp. 2290–2295.

[3] Z. Li and S. S. Sastry, "Task-oriented optimal grasping by multifingered robot hands," *IEEE Journal on Robotics and Automation*, vol. 4, no. 1, pp. 32–44, 1988.

[4] J. Mahler, F. T. Pokorny, B. Hou, M. Roderick, M. Laskey, M. Aubry, K. Kohlhoff, T. Kröger, J. Kuffner, and K. Goldberg, "Dex-net 1.0: A cloud-based network of 3d objects for robust grasp planning using a multi-armed bandit model with correlated rewards," in *Robotics and Automation (ICRA), 2016 IEEE International Conference on*. IEEE, 2016, pp. 1957–1964.

[5] K. Hang, J. A. Stork, and D. Kragic, "Hierarchical fingertip space for multi-fingered precision grasping," in *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*. IEEE, 2014, pp. 1641–1648.

[6] S. Levine, P. Pastor, A. Krizhevsky, and D. Quillen, "Learning hand-eye coordination for robotic grasping with large-scale data collection," in *International Symposium on Experimental Robotics*. Springer, 2016, pp. 173–184.

[7] L. Pinto and A. Gupta, "Supersizing self-supervision: Learning to grasp from 50k tries and 700 robot hours," in *Robotics and Automation (ICRA), 2016 IEEE International Conference on*. IEEE, 2016, pp. 3406–3413.

[8] M. Ciocarlie, C. Goldfeder, and P. Allen, "Dexterous grasping via eigengrasps: A low-dimensional approach to a high-complexity problem," in *Robotics: Science and Systems Manipulation Workshop-Sensing and Adapting to the Real World*. Citeseer, 2007.

[9] Y. Li, J. L. Fu, and N. S. Pollard, "Data-driven grasp synthesis using shape matching and task-based pruning," *IEEE Transactions on Visualization and Computer Graphics*, vol. 13, no. 4, pp. 732–747, 2007.

[10] E. Klingbeil, D. Rao, B. Carpenter, V. Ganapathi, A. Y. Ng, and O. Khatib, "Grasping with application to an autonomous checkout robot," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*. IEEE, 2011, pp. 2837–2844.

[11] A. ten Pas and R. Platt, "Using geometry to detect grasp poses in 3d point clouds," in *Robotics Research*. Springer, 2018, pp. 307–324.

[12] Y. Fan, H.-C. Lin, T. Tang, and M. Tomizuka, "Grasp planning for customized grippers by iterative surface fitting," *arXiv preprint arXiv:1803.11290*, 2018.

[13] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 580–587.

[14] Experimental Videos for Learning Grasp Exploration for Robust Bin Picking by RCNN-ISF, http://me.berkeley.edu/%7Eyongxiangfan/ICRA2019/rcnnisf.html.

[15] P. J. Besl and N. D. McKay, "Method for registration of 3-d shapes," in *Sensor Fusion IV: Control Paradigms and Data Structures*, vol. 1611. International Society for Optics and Photonics, 1992, pp. 586–607.

[16] A. Myronenko and X. Song, "Point set registration: Coherent point drift," *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, no. 12, pp. 2262–2275, 2010.

[17] F. L. Bookstein, "Principal warps: Thin-plate splines and the decomposition of deformations," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 11, no. 6, pp. 567–585, 1989.

[18] T. Jost and H. Hugli, "A multi-resolution icp with heuristic closest point search for fast and robust 3d registration of range images," in *3-D Digital Imaging and Modeling, 2003. 3DIM 2003. Proceedings. Fourth International Conference on*. IEEE, 2003, pp. 427–433.

[19] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *International Conference on Neural Information Processing Systems*, 2012, pp. 1097–1105.

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

[21] Y. Fan, T. Tang, H.-C. Lin, and M. Tomizuka, "Real-time grasp planning for multi-fingered hands by finger splitting," *arXiv preprint arXiv:1804.00050*, 2018.