

Collision-Aware Target-Driven Object Grasping in Constrained Environments

Xibai Lou¹, Yang Yang² and Changhyun Choi¹

Abstract— Grasping a novel target object in constrained environments (e.g., walls, bins, and shelves) requires intensive reasoning about grasp pose reachability to avoid collisions with the surrounding structures. Typical 6-DoF robotic grasping systems rely on the prior knowledge about the environment and intensive planning computation, which is ungeneralizable and inefficient. In contrast, we propose a novel Collision-Aware Reachability Predictor (CARP) for 6-DoF grasping systems. The CARP learns to estimate the collision-free probabilities for grasp poses and significantly improves grasping in challenging environments. The deep neural networks in our approach are trained fully by self-supervision in simulation. The experiments in both simulation and the real world show that our approach achieves more than 75% grasping rate on novel objects in various surrounding structures. The ablation study demonstrates the effectiveness of the CARP, which improves the 6-DoF grasping rate by 95.7%.

Index Terms— Grasping, Deep Learning in Grasping and Manipulation, Perception for Grasping and Manipulation

I. INTRODUCTION

Target-driven grasping is a fundamental yet challenging task in robotic manipulation, as it requires intensive reasoning about grasping stability from imperfect and partial observations. Most grasping systems assume a table-top scenario and simply choose 3-DoF grasp poses to mitigate the difficulty of reasoning. However, to grasp novel targets in constrained environments, an autonomous robot has to expand its action space from 3-DoF to 6-DoF, as shown in Fig. 1. In addition, these environments escalate two challenges: 1) How to robustly perceive novel target objects and surrounding structures? and 2) How to foresee the influence of surrounding structures on the grasping success probability?

In recent years, target-driven grasping approaches have been proposed by combining off-the-shelf object recognition modules (e.g., detection, template matching, and classifiers) with data-driven grasping models [1], [2]. These approaches focus on object-centric reasoning for grasping (i.e., predicting grasping stability from object appearance or geometry) while overlooking scene context beyond objects. As a result, in constrained environments with surrounding structures (e.g., walls, bins), they have to plan for an enormous set of sampled grasp poses and iteratively search through the entire set with a collision-checking algorithm. Furthermore,

*This work was in part supported by the MnDRIVE Initiative on Robotics, Sensors, and Advanced Manufacturing.

¹X. Lou and C. Choi are with the Department of Electrical and Computer Engineering, Univ. of Minnesota, Minneapolis, USA {lou00015, cchoi}@umn.edu

²Y. Yang is with the Department of Computer Science and Engineering, Univ. of Minnesota, Minneapolis, USA yang5276@umn.edu

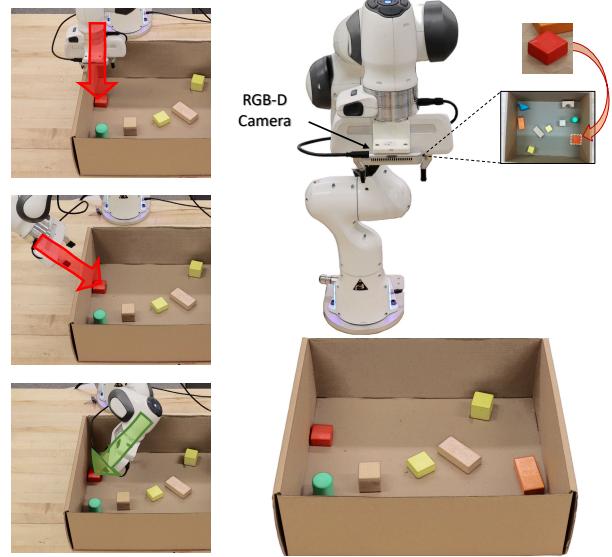


Fig. 1: Grasping a target object in constrained environments. Given a query image, our approach is able to localize and grasp the target object (red cuboid) surrounded by structures through reasoning feasible 6-DoF poses. In particular, our Collision-Aware Reachability Predictor (CARP) helps solve this challenging task by estimating the collision-free probability of each grasp pose.

these approaches require complete knowledge about the environment (e.g., geometric models of surrounding structure), which is, in practice, usually partially observable by imperfect sensors. Hence, these approaches suffer from excessive planning failures due to the absence of collision-awareness. Another limitation of these approaches lies in perception. Though simulation can expedite development and training, these RGB image-based object recognition modules require additional efforts to bridge the sim-to-real gap [3], [4] and generalize poorly to novel objects. These limitations motivate us to develop a target-driven grasping pipeline that achieves single-shot recognition for novel objects and requires only single planning for 6-DoF grasping in constrained environments.

The proposed collision-aware target-driven grasping pipeline integrates a robust perception module and a collision-aware 6-DoF grasping module. The perception module exploits the depth information in simulation with Siamese networks [5] for single-shot recognition and sim-to-real generalization. Our 6-DoF grasping module features a Collision-Aware Reachability Predictor (CARP). The CARP is a 3D convolutional neural network (3D CNN) that explicitly learns the probabilities of reaching a set of grasp poses without having collisions between the robot manipulator

and surrounding structures. The overall feasibility of 6-DoF grasp poses is evaluated by combining the collision-free probabilities and the predictions of grasping stability from a 3D CNN-based Grasp Stability Predictor (GSP) [6].¹

The rationale behind our approach resides in human behavior. For instance, when grasping an item from a packaging box, we naturally optimize our grasping action (in terms of both reachability and stability) with our experience rather than iterative checking. Our work is an early attempt to explicitly estimate collisions in constrained environments for target-driven 6-DoF grasping. The main contributions of our work are as follows:

- The Collision-Aware Reachability Predictor (CARP) that models the correlation between the spatial information and the collision-free probability of 6-DoF grasp poses with a 3D CNN. The CARP is trained with synthetic depth data by self-supervision and directly transferred to the real world.
- A target-driven robotic grasping pipeline that comprises a depth-based single-shot recognition module, the Collision-Aware Reachability Predictor, and the Grasp Stability Predictor. The pipeline localizes the novel target object in clutter and then grasps it in a constrained environment with single planning.

II. RELATED WORK

A. Target-driven Object Grasping

Robotic grasping is a fundamental but challenging problem in robotic manipulation [7], [8]. This vast literature can be divided into model-based [9], [10] and learning-based approaches [11], [12]. Another way of categorization is by task goal: target-agnostic [13], [14] and target-driven grasping [2], [15]. Our approach is learning-based and target-driven. Recent target-agnostic grasping approaches apply deep neural networks to learn grasping in 3-DoF action space (i.e., grasp pose with a 2D position and a wrist orientation) [13], [14], [16], [17], [18]. While 3D CNN was mainly studied in object recognition tasks [19], [20], Choi *et al.* [21] showed with a soft robot hand that it is capable of proposing additional grasping directions beyond the standard top-down poses. A few works learn to propose 6-DoF poses [6], [22], [23], [24], [25]. Mousavian *et al.* trained a network based on PointNet++ and a variational autoencoder [26] to generate stable grasp poses. Target-driven grasping [2], [27], [28], [29] is less studied compared to target-agnostic ones. The target-driven grasping problem necessitates a perception module, such as 2D image based template matching [30] and semantic segmentation [27], that recognizes the target object before grasping. Single-shot recognition using traditional RGB-based Siamese Networks [5] has also been explored for robotic grasping task [15]. Recent studies suggests that synthetic depth data is less influenced by the sim-to-real gap [11], [31]. We exploit the depth data both for object perception and grasping to minimize such sim-to-real gap.

¹Note that the acronym GSP refers to the 3D CNN module in [6] We did not explicitly use the GSP in [6] but defined here for a concise reference.

B. Grasping Reachability

Typical robotic grasping pipelines need to solve inverse kinematics with motion planning algorithms [32], [33] to reach a goal pose. Though such algorithms can handle reachability and collision during execution, the computational cost is remarkably high, especially when the trajectory to execute is infeasible. Most approaches bypass this problem by restricting target objects within a known reachable workspace [14], [21], [23]; other works estimate the grasping reachability by querying an offline [34], or online [35] database of reachable grasp poses. The Reachability Predictor in our prior work [6] only predicts the reachability concerning the kinematics of robot arms in the table-top scenario. Hence, it does not consider the collision-free probability, which determines the grasping reachability in the constrained environments. A comparable work from Murali *et al.* [29] extends their previous work [24] to estimate the collision score between object and gripper for 6-DoF grasp poses. We consider beyond gripper-object collisions as our collision sources further include surrounding structures.

III. PROBLEM FORMULATION

We consider the problem of generating feasible 6-DoF grasp poses for a target object surrounded by structures and other objects. The problem is formulated as follows:

Definition 1. A grasp pose $\mathbf{X} \in SE(3)$ is **collision-free** if the robot arm is able to reach the goal configuration without colliding with the surrounding structures.

Assumption 1. The target object is possibly unknown (i.e., novel objects) and partially observable (e.g., object occlusions and imperfect sensors), but an image of the target object is given as the only target information.

The scene point cloud \mathcal{P}_s , obtained from a single view RGB-D image \mathbf{I} , includes target object point cloud $\mathcal{P} \subset \mathbb{R}^3$ and a surrounding structure point cloud $\mathcal{P}' \subset \mathbb{R}^3$. To explore the full 6-DoF action space, we do not pose any constraint on the sampled grasp pose set \mathcal{X} . The detailed sampling procedure is described in [6].

Let $\mathcal{S}_c(\mathbf{X}) \in \{0, 1\}$ denote a binary-valued collision-free metric where $\mathcal{S}_c = 1$ indicates that the grasp is collision-free. The collision-free probability is determined solely by the spatial relationship between the manipulator and the surrounding structures, given by $p_c(\mathbf{X}, \mathcal{P}') = Pr(\mathcal{S}_c = 1 | \mathbf{X}, \mathcal{P}')$.

Each grasp pose $\mathbf{X} \in \mathcal{X}$ is also subject to a binary-valued stability metric $\mathcal{S}_g(\mathbf{X}) \in \{0, 1\}$ where $\mathcal{S}_g = 1$ indicates that the grasp pose is stable. We would like to estimate the grasping stability $p_g(\mathbf{X}, \mathcal{P}) = Pr(\mathcal{S}_g = 1 | \mathbf{X}, \mathcal{P})$. Finally, the feasibility metric $\mathcal{S}_f(\mathbf{X}) \in \{0, 1\}$ measures if the grasp pose is feasible to accomplish the task, and $\mathcal{S}_f = 1$ indicates a feasible, and therefore simultaneously stable and collision-free grasp pose. Note that the stability metric of a pose \mathbf{X} is independent of its collision-free metric. Thus, we consider the grasping feasibility $p_f(\mathbf{X}, \mathcal{P}_s)$ as the joint probability of the two independent probabilities $p_c(\mathbf{X}, \mathcal{P}')$ and $p_g(\mathbf{X}, \mathcal{P})$.

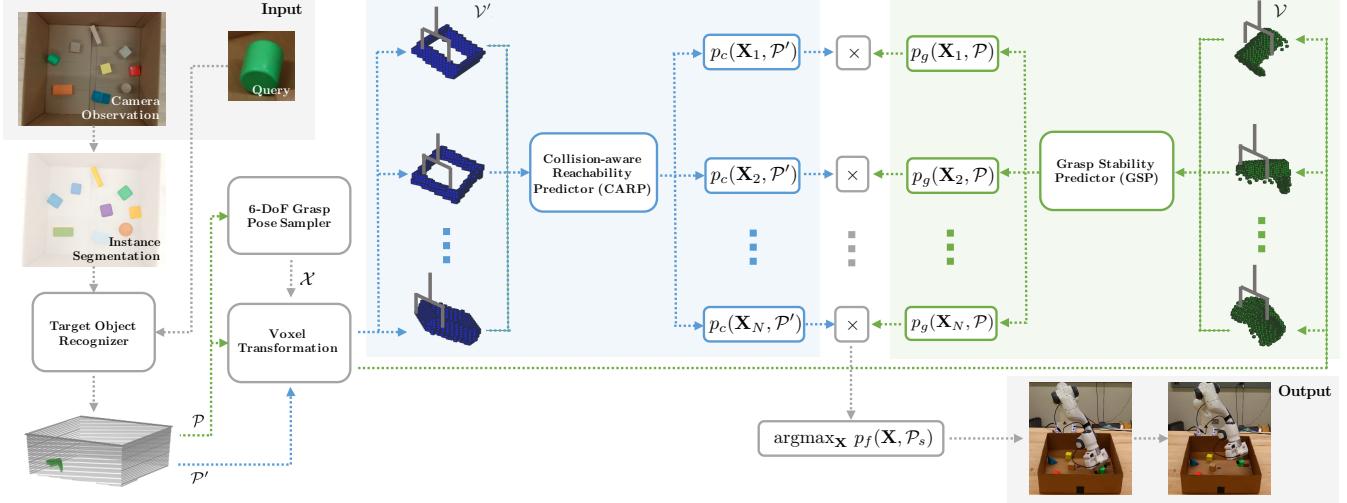


Fig. 2: Grasping pipeline. The scene point cloud \mathcal{P}_s is first reconstructed from RGB-D image \mathbf{I} . Then the perception module takes \mathbf{I} as input and gives the target object point cloud \mathcal{P} and the surrounding structure cloud \mathcal{P}' . From \mathcal{P} , a set of 6-DoF grasp poses \mathcal{X} is randomly sampled, and \mathcal{P}' is transformed by each $\mathbf{X} \in \mathcal{X}$ and voxelized to voxel grid \mathcal{V}' . The CARP takes \mathcal{V}' as input and evaluates the collision-free probability p_c . The voxel grid \mathcal{V} is transformed from the object point cloud \mathcal{P} w.r.t \mathbf{X} . The Grasp Stability Predictor then evaluates the grasping stability p_g for each \mathcal{V} . We multiply p_c and p_g to get a grasping feasibility p_f and execute the pose having the highest p_f .

IV. PROPOSED APPROACH

We propose a 6-DoF target-driven pipeline for object grasping in constrained environments. Our approach aims to 1) recognize a target object (seen or novel) with a partial observation and 2) select the most feasible grasp pose from a set of randomly sampled 6-DoF grasp poses.

A. System Overview

As illustrated in Fig. 2, our grasping pipeline comprises a perception module and a grasping module. Given a query image for the target object, the perception module first separates the scene point cloud \mathcal{P}_s into target point cloud \mathcal{P} and surrounding structures \mathcal{P}' . The point clouds are then forwarded to our grasping module, which consists of the Collision-Aware Reachability Predictor (CARP) and the Grasp Stability Predictor (GSP). The CARP evaluates the collision-free probability $p_c(\mathbf{X}, \mathcal{P}')$ for grasp pose candidates \mathbf{X} by using the structure point cloud \mathcal{P}' . The CARP takes advantage of the spatial relationship between the robot hand and all the surrounding structures to determine if the given

pose is prone to collision with the structure. The GSP evaluates the stability of the grasp poses on the object point cloud \mathcal{P} . The final grasp pose is selected to maximize the overall grasping feasibility $p_f(\mathbf{X}, \mathcal{P}_s)$. Our grasping pipeline can be seen as a combination of a peripheral vision (i.e., the CARP's coarse but wide-angle understanding of the surrounding structures) and a foveated vision (i.e., the GSP's finer yet narrow-viewed understanding of the target object).

B. Perception

The perception module is delineated in Algorithm 1 wherein it first takes as input a scene RGB-D image \mathbf{I} from the camera and generates a set of binary-valued class-agnostic instance masks $\mathcal{M}_{1,\dots,N} \in \mathbb{Z}^{H \times W}$ by SD Mask R-CNN [31]. Next, the set of the masks is multiplied with the RGB-D image to generate object images for recognition, where a queried RGB-D image of the target object is supplied. Unlike a traditional RGB image-based Siamese network [5], our approach includes the depth channel that helps differentiate the challenging objects (e.g., having similar visual appearance but different geometric shapes) and improves the sim-to-real generalization. To achieve single-shot recognition, our RGB-D Siamese CNN first extracts the latent features of an object image and the queried image for feature matching. During training, the L1-distance between the two feature vectors is minimized for the same class objects using a contrastive loss [36]. Therefore, the best-matched object during testing will have the lowest distance in a forward pass and is selected as the target object. The masked depth image is used to reconstruct the 3D target point cloud \mathcal{P} through back-projection while the non-masked region gives the structure point cloud \mathcal{P}' .

Algorithm 1 Perception Module

Input: Scene RGB-D Image \mathbf{I} , Query RGB-D Image \mathbf{I}_q , SD Mask-RCNN \mathcal{N}_s , Siamese Network \mathcal{N}_r
Output: Target Cloud \mathcal{P} , Structure Cloud \mathcal{P}'

- 1: $\mathcal{M} \leftarrow \mathcal{N}_s.\text{InstanceSegmentation}(\mathbf{I})$
- 2: **for** $\mathbf{M} \in \mathcal{M}$ **do**
- 3: $\mathbf{I}_o \leftarrow \text{Mask}(\mathbf{I}, \mathbf{M})$
- 4: $s(\mathbf{I}_o, \mathbf{M}) \leftarrow \mathcal{N}_r.\text{Recognition}(\mathbf{I}_o, \mathbf{I}_q)$
- 5: $\mathbf{I}_t, \mathbf{M}_t \leftarrow \text{argmax}_{\mathbf{I}_o} s(\mathbf{I}_o, \mathbf{M})$
- 6: $\mathbf{I}_s \leftarrow \mathbf{I} \cap \overline{\mathbf{M}_t}$
- 7: $\mathcal{P}, \mathcal{P}' \leftarrow \text{BackProjection}(\mathbf{I}_t, \mathbf{I}_s)$

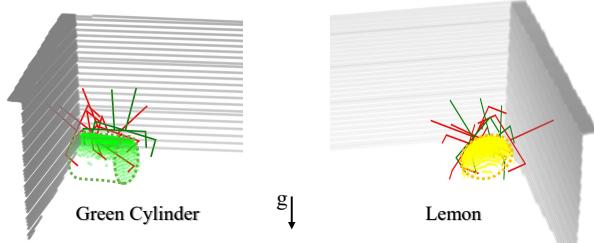


Fig. 3: **The CARP predictions.** The target object is placed adjacent to the surrounding structures. The CARP predicts the collision-free probabilities of each grasp pose. The green poses have higher results and are less prone to collision.

C. Grasping

Our grasping module first considers the entire environmental structures to capture all potential collisions. It encodes the spatial information by transforming structure cloud \mathcal{P} with respect to the grasp pose \mathbf{X} . Then it voxelizes the transformed \mathcal{P}' to a $40 \times 40 \times 40$ binary voxel occupancy grid \mathcal{V}' , whose voxel size is $(0.025m)^3$. It is centered on the grasp point \mathbf{p} , and its coordinate frame is aligned with that of the grasp pose \mathbf{X} . Consequently, collision-free and colliding poses correspond to \mathcal{V}' with distinct characteristics, and the CARP learns from these features to estimate the collision-free probability p_c from the input voxel. The spatial features are extracted by the 3D CNNs and then fed to fully connected layers. The output layer uses a sigmoid activation function to model the collision-free probability of \mathbf{X} . Since a given grasp pose can either be 0 (collision) or 1 (collision-free), we use binary cross-entropy as the loss function.

To estimate the grasping stability for 6-DoF poses, we use a 3D CNN-based Grasp Stability Predictor (GSP), details of implementation can be found in [6]. It focuses on the target object and examines the geometric shape. Moreover, it implicitly minimizes collisions with clutters by evaluating a more informative update of \mathcal{P} , obtained by center cropping the scene point cloud \mathcal{P}_s at a grasping point \mathbf{p} by $(0.1m)^3$. The new \mathcal{P} is transformed with respect to the grasp pose \mathbf{X} and then voxelized as the input \mathcal{V} to the GSP. Since

Algorithm 2 Collision-Aware Target-driven Grasping

Input: RGB-D image \mathbf{I} , GSP \mathcal{N}_g , CARP \mathcal{N}_c
Output: collision-free grasp pose $\mathbf{X}_f \in SE(3)$

- 1: $\mathcal{P}_s \leftarrow \text{BackProjection}(\mathbf{I})$
- 2: $\mathcal{P}, \mathcal{P}' \leftarrow \text{Perception}(\mathbf{I})$
- 3: $\mathcal{X} \leftarrow \text{GraspPoseSampling}(\mathcal{P})$
- 4: **for** $\mathbf{X} \in \mathcal{X}$ **do**
- 5: $\mathcal{P} \leftarrow \text{CenterCropping}(\mathcal{P}_s, \mathbf{X})$
- 6: $\mathcal{V}, \mathcal{V}' \leftarrow \text{VoxelTransformation}(\mathcal{P}, \mathcal{P}', \mathbf{X})$
- 7: $p_c \leftarrow \mathcal{N}_c.\text{Feedforward}(\mathcal{V}')$
- 8: $p_g \leftarrow \mathcal{N}_g.\text{Feedforward}(\mathcal{V})$
- 9: $p_f \leftarrow p_c \times p_g$
- 10: $\mathbf{X}_f \leftarrow \text{argmax}_{\mathbf{X} \in \mathcal{X}} p_f(\mathbf{X}, \mathcal{P}_s)$
- 11: **Grasp**(\mathbf{X}_f)

\mathcal{V} is voxelized from the cropped point cloud \mathcal{P} , it may partially contain the voxels of adjacent objects, which tend to lower grasping stability predictions as these voxels make the geometric shape in \mathcal{V} unfamiliar. Therefore, the GSP will pick the pose that contains the least collision with other surrounding objects. We choose the most feasible pose as the final grasp pose to be executed, which is simultaneously collision-free and stable. Algorithm 2 shows the flow of our grasping system.

D. Data Collection and Training

The entire system is trained by self-supervision in simulation. We first train the perception module with 500 RGB-D images of each object with ground truth class labels (accessible in simulation). For grasping module, we generated the training dataset by dropping the training objects within a common workspace structure (e.g., wall or bin) of random size (ranging from $0.3m$ to $0.5m$) and orientation. Then the robot interacts with the objects and collects 60,000 labeled point clouds to train the CARP. The labels are generated reliably and efficiently with the default collision checking algorithm in the simulation environment, which assumes the full knowledge of the scene. To decouple collision from grasping results, a training dataset of 50,000 data is collected separately to train the GSP in the table-top scenario where no surrounding structures exist.

V. EXPERIMENTS

We evaluate our approach in both simulated and real-world settings. The experiments are designed to answer three questions: 1) Can the perception module robustly identify the target object, including novel ones, given a query image?, 2) Can the CARP improve planning efficiency?, and 3) How does our approach perform compared to other state-of-the-art grasping approaches in various testing scenarios?

Baselines: We first performed ablation studies on the perception module. The performance of the pipeline is then compared with 4 baseline methods: 1) **RAND** randomly generates a grasp pose on the target object, 2) **6GN** is 6-DoF GraspNet [24] that learns to generate 6-DoF grasp poses with PointNet++[37] and a variational autoencoder, 3) **VPG** [14] learns both pushing and grasping from RGB-D images, and we only evaluate the grasping part of this work by filtering the grasp Q-map with a target object mask, and 4) **GSP+RP** combines the Grasp Stability Predictor (GSP) with the Reachability Predictor (RP) [6] that only considers the kinematic constraints of a robot arm when proposing feasible 6-DoF grasp poses. Note that the mask output from our perception module is input to each baseline as they were originally designed for target-agnostic grasping.

Evaluation Metrics: We define two metrics for both simulation and real-world evaluations, the planning rate and the grasping rate. The first metric is defined as the planning rate = $\frac{\# \text{ of successful plans}}{\# \text{ of total trials}}$. A motion planning is considered successful only if the motion planning algorithm is able to find a valid trajectory for the robot arm without any collision. For grasping, the grasping rate is defined as $\frac{\# \text{ of successful grasps}}{\# \text{ of proposed grasps}}$.

TABLE I: Target Recognition Accuracy

	Known	Novel	Occl.	Novel-Occl.
RGB Siamese	78.6	63.3	73.6	54.4
RGB-D Siamese (Ours)	96.3	85.5	93.5	80.4

A grasp is successful only if the robot gripper successfully reaches the object and lifts the object by 15 cm. Because the performance of each method varies depending on the object arrangement (i.e., an object is much more difficult to grasp if it is close to surrounding structures), we design a standard and a challenging arrangement to demonstrate the effectiveness of the CARP, shown in Fig. 4. If a target object is placed at an unreachable location (e.g., target objects fall at corners of the bin), we rearrange the object and continue the experiment.

A. Perception Experiments

The perception module is tested on known and novel objects that are randomly dropped into a bin. Note the novel objects include challenging ones that are similar in color but different in shape, as shown in Fig. 9, which is non-trivial for RGB-based perception. We further test the robustness of our perception module against occlusion by hiding 25% of the input images. Table I summarizes the performance of different perception modules using a small fine-tune dataset of 500 and a test dataset of 1000 image pairs. The results show the effectiveness of depth information. Our perception module outperforms the RGB only version by 35.07% on novel objects and is robust to occlusion.

B. Simulation Experiments

Our simulation environment is built in CoppeliaSim [38] with Bullet [39] physics engine 2.83. The simulation setup uses a Panda robot arm, various workspace structures, and different test objects. We choose an eye-on-hand camera configuration to expand the field of view and minimize occlusion from surrounding structures.

In the standard arrangements, we use different workspace structures, such as walls, large boxes, and small boxes, and randomly orient them, as shown in Fig. 5. Ten objects are randomly dropped in the environment, and then we execute each approach 31 times. The target object is dropped into

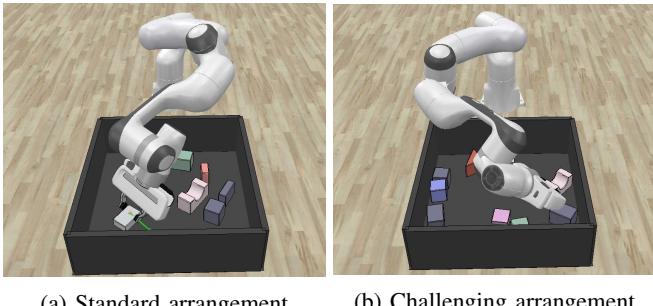


Fig. 4: **Different arrangements in simulation.** The standard arrangement in (a) test our approach in a common settings while the challenging arrangement in (b) reflects a manually designed scenario where collision-awareness is even more critical.

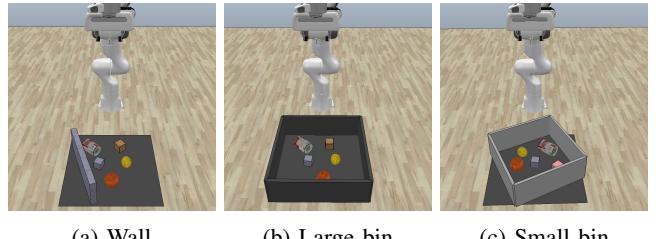


Fig. 5: **Different structures in simulation.** Our approach is able to grasp a target object in various surrounding structures with 6-DoF poses. As the testing environment gets increasingly challenging, the effectiveness of the CARP becomes more obvious.

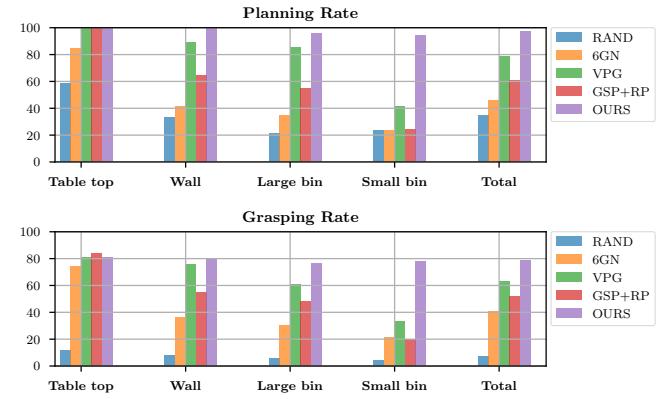


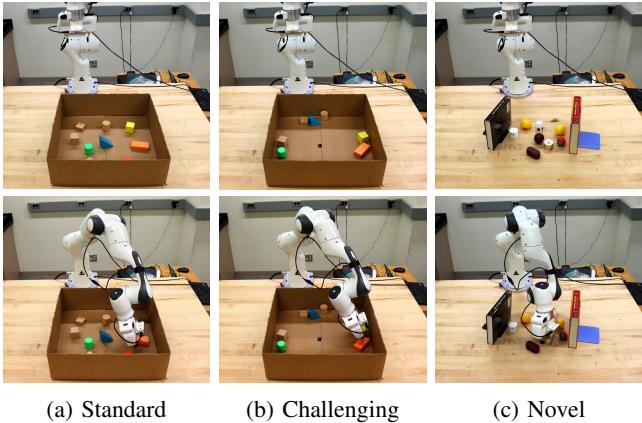
Fig. 6: **Performance in simulation.** The planning rate (top) and grasping rate (bottom) of each approach in various surrounding structures. Our approach achieves 97.55% planning rate and 78.78% grasping rate on average, suggesting that it is the most effective.

the workspace after a successful grasp to keep the test objects consistent. We report the results of each method in the standard arrangement in Fig. 6. We noticed that as the workspace's constraints become increasingly stringent, other approaches experience notable planning difficulties due to collision, while the planning rate of our approach degrades much less with the help of the CARP, hence increasing both the planning and grasping efficiencies. On average, our approach is able to achieve a 97.55% planning rate and a 78.78% grasping rate, which outperforms the baselines by large margins. The **GSP+RP** serves as an ablation baseline, introducing collision-awareness with the CARP increases both the planning and grasping rate of **GSP+RP** by more than 60%. We notice that 6-DoF GraspNet suffers from infeasible grasp poses. Although **VPG** manages to avoid some of the planning challenges by using top-down grasping exclusively, it fails both planning and grasping tasks when the target object is close to the workspace structures (e.g., Small bin).

In the challenging arrangement, the objects are manually dropped close to the peripheral of the bin. We evaluate the methods and summarize the results in Table II. The evaluation of the challenging arrangement requires better reasoning about surrounding structures for grasping. Our approach outperforms the best-performing baseline by 127.31% in grasping rate, showing that **OURS** is especially effective under this scenario.

TABLE II: Challenging Arrangement in Simulation

	RAND	VPG	6GN	GSP+RP	OURS
Planning Rate	12.90	41.94	21.57	26.67	93.55
Grasping Rate	3.23	35.48	20.46	23.33	80.65



(a) Standard (b) Challenging (c) Novel

Fig. 7: **Examples of real world experiments.** Our approach is able to grasp a target object that is close to the wall with 6-DoF grasp poses. This task is challenging as any small variation in grasp pose may lead to a collision with the surrounding structures.

C. Real-robot Experiments

We further evaluate our approach and the baselines on a Franka Emika Panda robot. A single-view RGB-D image is taken with an Intel Realsense D435 camera in eye-on-hand configuration. For a given pose, the robot follows the corresponding trajectory generated with MoveIt in open-loop. Fig. 7 shows our real-robot experiment settings, which include standard, challenging, and novel arrangements of objects in the real world. Following the same evaluation metrics, we compare our approach with **RAND**, **6GN**, **VPG**, and **GSP+RP**. Fig. 8 compiles the performance of each method in different scenarios for 31 runs.

Overall, our approach is able to perform consistently well in the real world and achieves the highest planning rate and grasping rate. Both **GSP+RP** and **VPG** are prone to predicting unreachable poses because they overlook the other environment information, which contributes to failed grasps. During testing, we realized the benefits as well as the limitations of our approach. The voxel grid of the CARP has a fixed resolution², and thus unable to capture smaller variations of either grasp poses or point cloud. In the very challenging cases where small variations can influence the collision-free probability (e.g., target object located side-by-side with the wall), the resolution limitation on the voxel degrades the performance.

Our approach generalizes well to novel environments. We test our system with a random collection of novel objects, as shown in Fig. 9. Novel experiment results in Fig. 8 suggest our system is able to grasp novel objects as well. Note that there is no further training in both our perception and

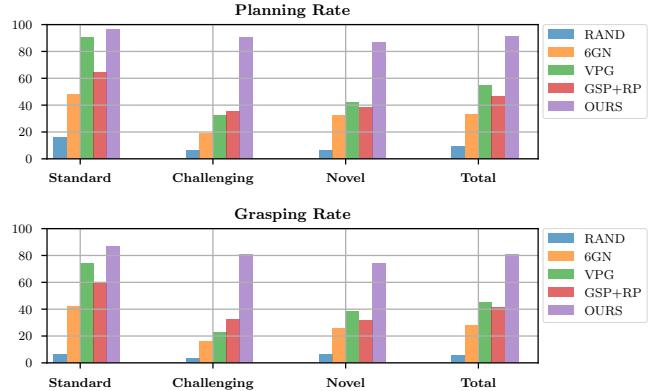


Fig. 8: **Performance in the real world.** The planning rate (top) and grasping rate (bottom) of each approach in different arrangements. Our approach achieves 91.40% planning rate and 80.65% grasping rate on average, corroborating the simulation results.

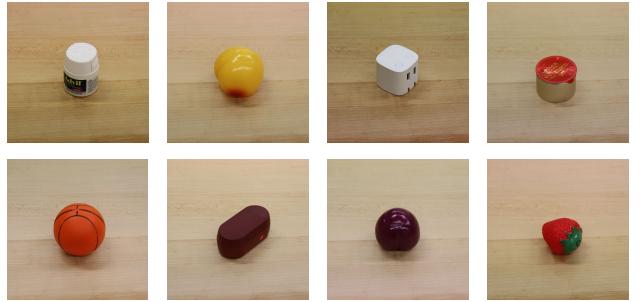


Fig. 9: **Examples of novel objects.** We show that our work is able to generalize to objects that were not seen during training.

grasping algorithms as the perception pipeline generalizes to novel objects thanks to the Siamese network and the 3D CNN-based CARP and GSP, although additional fine-tuning would further improve the accuracy of our system. As the point clouds are transformed to gripper coordinates, changing the robot’s gripper pose will subsequently modify its coordinate system and leave the performance of the system intact. The CARP is implicitly conditioned on robot hardware, therefore, it may require additional tuning in order to generalize to other robot manipulators.

VI. CONCLUSION

In this work, we presented the Collision-Aware Reachability Predictor (CARP), a learning-based approach that is able to accurately estimate collisions between the robot arm and surrounding structures using spatial information. Simulated and real experiments in various scenarios clearly showed the benefit of using the CARP in terms of planning rate and grasping rate. We further proposed a grasping pipeline that integrated our perception module and grasping module, and achieved, on average, a 91.40% planning rate and a 80.65% grasping rate in real-robot experiments. The proposed approach outperformed the other baseline methods by large margins. As future work, it would be interesting to include robot hand shapes in learning, so that our approach could better generalize to different robot hands.

²One voxel of the CARP covers 2.5^3cm^3 of real robot workspace.

REFERENCES

- [1] K. Fang, Y. Bai, S. Hinterstoisser, S. Savarese, and M. Kalakrishnan, “Multi-task domain adaptation for deep learning of instance grasping from simulation,” in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 3516–3523.
- [2] Q. Lin and D. Chen, “Target recognition and optimal grasping based on deep learning,” in *2018 WRC Symposium on Advanced Robotics and Automation (WRC SARA)*, 2018, pp. 1–6.
- [3] Y. Chebotar, A. Handa, V. Makovychuk, M. Macklin, J. Issac, N. Ratliff, and D. Fox, “Closing the sim-to-real loop: Adapting simulation randomization with real world experience,” in *2019 International Conference on Robotics and Automation (ICRA)*, 2019, pp. 8973–8979.
- [4] K. Bousmalis, A. Irpan, P. Wohlhart, Y. Bai, M. Kelcey, M. Kalakrishnan, L. Downs, J. Ibarz, P. Pastor, K. Konolige, et al., “Using simulation and domain adaptation to improve efficiency of deep robotic grasping,” in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 4243–4250.
- [5] G. Koch, R. Zemel, and R. Salakhutdinov, “Siamese neural networks for one-shot image recognition,” in *ICML deep learning workshop*, vol. 2. Lille, 2015.
- [6] X. Lou, Y. Yang, and C. Choi, “Learning to generate 6-dof grasp poses with reachability awareness,” in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 1532–1538.
- [7] A. Sahbani, S. El-Khoury, and P. Bidaud, “An overview of 3d object grasp synthesis algorithms,” *Robotics and Autonomous Systems*, vol. 60, no. 3, pp. 326–336, 2012.
- [8] J. Bohg, A. Morales, T. Asfour, and D. Kragic, “Data-driven grasp synthesis—a survey,” *IEEE Transactions on Robotics*, vol. 30, no. 2, pp. 289–309, 2013.
- [9] A. Miller and P. Allen, “Graspit! a versatile simulator for robotic grasping,” *IEEE Robotics & Automation Magazine*, vol. 11, no. 4, pp. 110–122, 2004.
- [10] J. Weisz and P. K. Allen, “Pose error robust grasping from contact wrench space metrics,” in *2012 IEEE international conference on robotics and automation*. IEEE, 2012, pp. 557–562.
- [11] J. Mahler, J. Liang, S. Niyaz, M. Laskey, R. Doan, X. Liu, J. A. Ojea, and K. Goldberg, “Dex-net 2.0: Deep learning to plan robust grasps with synthetic point clouds and analytic grasp metrics,” *Robotics: Science and Systems (RSS)*, 2017.
- [12] 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*, vol. 37, no. 4–5, pp. 421–436, 2018.
- [13] L. Pinto and A. Gupta, “Supersizing self-supervision: Learning to grasp from 50k tries and 700 robot hours,” in *2016 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2016, pp. 3406–3413.
- [14] A. Zeng, S. Song, S. Welker, J. Lee, A. Rodriguez, and T. Funkhouser, “Learning synergies between pushing and grasping with self-supervised deep reinforcement learning,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018.
- [15] M. Danielczuk, A. Kurenkov, A. Balakrishna, M. Matl, D. Wang, R. Martín-Martín, A. Garg, S. Savarese, and K. Goldberg, “Mechanical search: Multi-step retrieval of a target object occluded by clutter,” in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, 2019.
- [16] H. Liang, X. Lou, and C. Choi, “Knowledge induced deep q-network for a slide-to-wall object grasping,” *arXiv preprint arXiv:1910.03781*, 2019.
- [17] 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.
- [18] D. Kappler, J. Bohg, and S. Schaal, “Leveraging big data for grasp planning,” *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 4304–4311, 2015.
- [19] D. Maturana and S. Scherer, “Voxnet: A 3d convolutional neural network for real-time object recognition,” in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2015, pp. 922–928.
- [20] S. Song, F. Yu, A. Zeng, A. X. Chang, M. Savva, and T. Funkhouser, “Semantic scene completion from a single depth image,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 1746–1754.
- [21] C. Choi, W. Schwarting, J. DelPreto, and D. Rus, “Learning object grasping for soft robot hands,” *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 2370–2377, 2018.
- [22] M. Gualtieri, A. Ten Pas, K. Saenko, and R. Platt, “High precision grasp pose detection in dense clutter,” in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 598–605.
- [23] A. ten Pas, M. Gualtieri, K. Saenko, and R. Platt, “Grasp pose detection in point clouds,” *The International Journal of Robotics Research*, vol. 36, no. 13–14, pp. 1455–1473, 2017.
- [24] A. Mousavian, C. Eppner, and D. Fox, “6-dof grapsnet: Variational grasp generation for object manipulation,” in *International Conference on Computer Vision (ICCV)*, 2019.
- [25] S. Song, A. Zeng, J. Lee, and T. Funkhouser, “Grasping in the wild: Learning 6dof closed-loop grasping from low-cost demonstrations,” *Robotics and Automation Letters (RA-L)*, 2020.
- [26] D. P. Kingma and M. Welling, “Auto-Encoding Variational Bayes,” in *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14–16, 2014, Conference Track Proceedings*, 2014.
- [27] Y. Yang, H. Liang, and C. Choi, “A deep learning approach to grasping the invisible,” *IEEE Robotics and Automation Letters*, 2020.
- [28] 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,” *The International Journal of Robotics Research*, vol. 39, no. 2–3, pp. 202–216, 2020.
- [29] A. Murali, A. Mousavian, C. Eppner, C. Paxton, and D. Fox, “6-dof grasping for target-driven object manipulation in clutter,” in *International Conference on Robotics and Automation (ICRA)*, 2020.
- [30] K. Fang, Y. Bai, S. Hinterstoisser, S. Savarese, and M. Kalakrishnan, “Multi-task domain adaptation for deep learning of instance grasping from simulation,” in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 2018, pp. 3516–3523.
- [31] M. Danielczuk, M. Matl, S. Gupta, A. Li, A. Lee, J. Mahler, and K. Goldberg, “Segmenting unknown 3d objects from real depth images using mask r-cnn trained on synthetic data,” in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 7283–7290.
- [32] S. Lavalle and J. Kuffner, “Rapidly-exploring random trees: Progress and prospects,” *Algorithmic and computational robotics: New directions*, 01 2000.
- [33] J. J. Kuffner Jr and S. M. LaValle, “Rrt-connect: An efficient approach to single-query path planning,” in *ICRA*, vol. 2, 2000.
- [34] O. Porges, T. Stouraitis, C. Borst, and M. A. Roa, “Reachability and capability analysis for manipulation tasks,” in *ROBOT2013: First Iberian Robotics Conference*, M. A. Armada, A. Sanfeliu, and M. Ferre, Eds. Cham: Springer International Publishing, 2014, pp. 703–718.
- [35] I. Akinola, J. Varley, B. Chen, and P. K. Allen, “Workspace aware online grasp planning,” 2018.
- [36] R. Hadsell, S. Chopra, and Y. LeCun, “Dimensionality reduction by learning an invariant mapping,” in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’06)*, vol. 2. IEEE, 2006, pp. 1735–1742.
- [37] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, “Pointnet: Deep learning on point sets for 3d classification and segmentation,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 652–660.
- [38] E. Rohmer, S. P. Singh, and M. Freese, “V-rep: A versatile and scalable robot simulation framework,” in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2013, pp. 1321–1326.
- [39] E. Coumans, “Bullet physics simulation,” in *ACM SIGGRAPH 2015 Courses*, ser. SIGGRAPH ’15. New York, NY, USA: ACM, 2015. [Online]. Available: <http://doi.acm.org/10.1145/2776880.2792704>