S^4G: Amodal Single-view Single-Shot $\mathbb{SE}(3)$ Grasp Detection in Cluttered Scenes

Yuzhe Qin[*1] **Rui Chen**[*1,2] **Hao Zhu**[1] **Meng Song**[1] **Jing Xu**[2]

Hao Su[1]
[1] University of California, San Diego
{y1qin, haozhu, mengsong, haosu}@eng.ucsd.edu
[2] Tsinghua University
chenr17@mails.tsinghua.edu.cn, jingxu@tsinghua.edu.cn

Abstract: Grasping is among the most fundamental and long-lasting problems in robotics study. This paper studies the problem of 6-DoF(degree of freedom) grasping by a parallel gripper in a cluttered scene captured using a commodity depth sensor from a single viewpoint. We address the problem in a learning-based framework. At the high level, we rely on a single-shot grasp proposal network, trained with synthetic data and tested in real-world scenarios. Our single-shot neural network architecture can predict amodal grasp proposal efficiently and effectively. Our training data synthesis pipeline can generate scenes of complex object configuration and leverage an innovative gripper contact model to create dense and high-quality grasp annotations. Experiments in synthetic and real environments have demonstrated that the proposed approach can outperform state-of-the-arts by a large margin.

Keywords: object grasping, single-shot grasp proposal, synthesis to real

1 Introduction

Grasping is among the most fundamental and long-lasting problems in robotics study. While classical model-based methods using mechanical analysis tools [1, 2, 3] can already grasp objects of known geometry, it remains an open problem of how to grasp generic objects in complex scenes.

Recently, data-driven approaches have shed light to addressing the generic grasp problem using machine learning tools [4, 5, 6, 7]. In order to readily generalize to unseen objects and layouts, a large body of recent works have focused on solving 3/4 DoF(degree of freedom) grasping, where the gripper is forced to approach objects from above vertically [8, 9]. Although this has greatly simplified the problem for picking and placing tasks, it has also inevitably restricted ways to interact with objects. For example, such grasping is unable to grab a horizontally placed plate. Worse still, top-down grasping often encounters difficulties in cluttered scenes with casually heaped objects, which requires extra hand freedoms for grasping buried objects. The limitation of 3/4 DoF grippers thus motivates the study of 6-DoF grippers to approach the object from arbitrary directions. We note that 6-DoF end-effector is essential to allow dexterous object manipulation tasks [10, 11].

This paper studies the 6-DoF grasping problem in a realistic yet challenging setting, assuming that a set of household objects from unknown categories are casually scattered on a table. A commodity depth camera is mounted with a fixed pose to capture this scene from only a single viewpoint, which gives a partial point cloud of the scene. The grasp is performed by a parallel gripper.

The setting is highly challenging for both perception and planning: First, the scene clutters limit viable grasp poses and may even fail the motion planning algorithms to achieve certain grasps. This challenge keeps us from considering 3/4-DoF grasp detection and restricts us to the more powerful yet sophisticated 6-DoF detection approach. Second, we make no assumptions of object categories. This *open set* setting puts us in a different category from existing semantic grasping method, such

[*] Equal contribution.

3rd Conference on Robot Learning (CoRL 2019), Osaka, Japan.

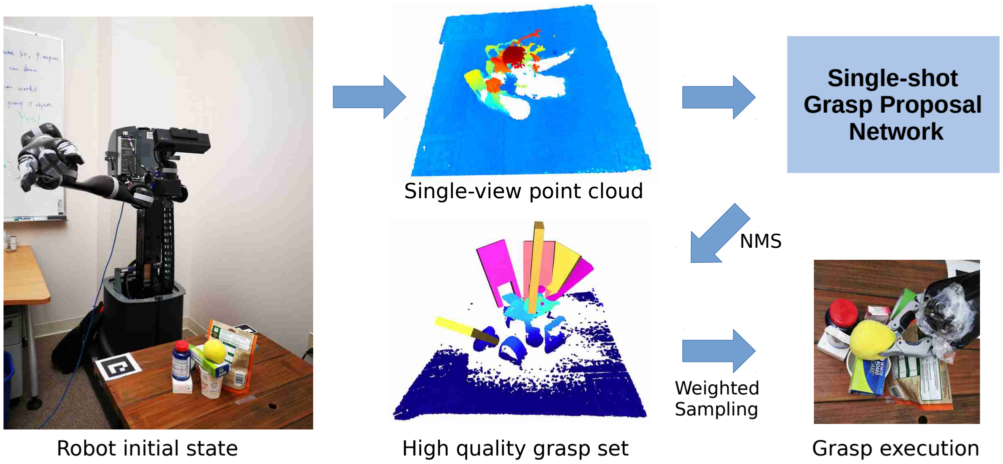


Figure 1: Illustration of the pipeline of Single-Shot SE(3) Grasp Detection (S^{4}G). Taking as input the view point cloud from the depth sensor, S^{4}G regresses the 6-DoF grasp pose directly and predicts the grasp quality for each point, which is more robust and effective. A set of high quality grasps are chosen, from which one grasp is sampled and executed.

as DexNet [12]. We require higher-level of generalizability based on better representation of the perceived content. Most existing methods can only work in simpler scenarios, by introducing high-quality and expensive 3D sensors for accurate scene capturing, or sensing the complete environment with multiple cameras[10], or assuming a scene of only a single object[13]. This challenge demands that our grasp detection has to be noise-resistant and *amodal*, i.e., being able to make an educated guess of the viable grasp from only a partial point cloud.

We address the challenges in a learning-based framework. At the high level, we rely on a single-shot grasp proposal network, trained with synthetic data and tested in real-world scenarios. Our design involves (1) a single-shot neural network architecture for amodal grasp proposal; and (2) a scene-level training data synthesis pipeline leveraging an innovative gripper contact model.

By its single-shot nature, our grasp proposal network enjoys better efficiency and accuracy compared with existing deep networks in the 6-DoF grasping literature. Existing work, such as [10], samples grasp candidates from $\mathbb{SE}(3)$ following some heuristics and assess their quality using networks. However, the running time goes up quickly as the number of sampled grasps increases, which makes the grasp optimization too slow. Unlike these approaches, we propose to directly *regress* 6-DoF grasps from the entire scene point cloud in one pass. Specifically, We are the first to propose a per-point scoring and pose regression method for 6-DoF grasp.

3D data from low-cost commercial depth sensors are partial, noisy and corrupted. To handle the imperfection of input 3D data, S^{4}G is trained by hallucinated point clouds of similar patterns, and it learns to extract robust features for grasp prediction from the corrupted data. We propose a simple yet effective gripper contact model to generate good grasps and associate these grasps to the point cloud. At inference time, we select high quality grasps based on the proposals of the network. Note that we are the first to generate a synthetic scene of many objects, rather than a single object, in the 6-DoF grasping literature.

The core novel insight of our S^{4}G is that we learn to propose possible grasps in this space by regression. We believe learning to regress grasp proposals would be the trend: For another problem of similar setting, object detection, the community has evolved from sliding windows to learning to generate object proposals. A second novelty is that, instead of generating training data by scenes of only a single object, we include multiple objects in the scene, with grasp proposals analyzed using a gripper contact model that considers touching area shape and size.

2 Related work

Deep Learning based Grasping Methods Caldera et al. [14] gave a thorough survey of deep learning methods for robotic grasping, which demonstrates the effectiveness of deep learning on

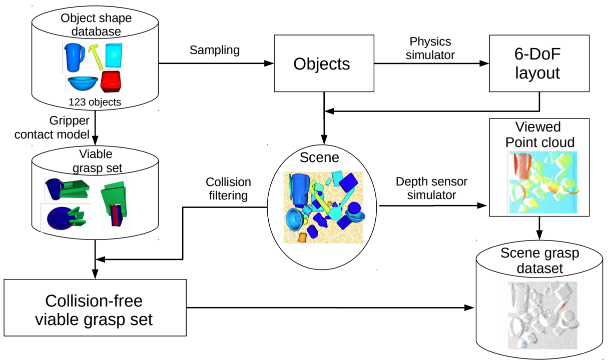


Figure 2: Flowchart of scene grasp dataset generation.

this task. In our paper, we focus on the problem of 6-DoF grasp proposal. Collet et al. [15], Zeng et al. [16], Mousavian et al. [17] tackled this problem by fitting the object model to the scan point cloud to retrieve the 6-DoF pose. Although it has shown promising results in industrial applications, the feasibility is limited in generic robotic application scenarios, e.g. house-holding robots, where the exact 3D models of numerous objects are not accessible. ten Pas et al. [10] proposed to generate grasp hypotheses only based on local geometry prior and attained better generalizability on novel objects, which was further extended by Liang et al. [18] by replacing multi-view projection features with direct point cloud representation. Because potential viable 6-DoF grasp poses are infinite, these methods guide the sampling process by constructing a Darboux frame aligned with the estimated surface normal and principal curvature and searching in its 6D neighbourhood. However, they may fail finding feasible grasps for thin structures, such as plates or bowls, where computing normals analytically from partial and noisy observation is challenging. In contrast to these sampling approaches, our framework is a single-shot grasp proposal framework [19, 14]–a direct regression approach for predicting viable grasp frames–which could handle flawed input well due to the network's knowledge. Moreover, by jointly analyzing local and global geometry information, our method not only considers the object of interest, but also its surroundings, which allows the generation of collision-free grasps in dense clutters.

Training Data Synthesis for Grasping Deep learning methods require an enormous volume of labelled data for the training process [9], however manually annotating 6-DoF grasp poses is not practical. Therefore, analytic grasp synthesis [20] is indispensable for ground truth data generation. These advanced models have provided guaranteed measurements of grasp properties with the availability of complete and precise geometric models of objects. In practice, the observation from sensors are partial and noisy, which undermines the metric accuracy. In the service of our single-shot grasp detection framework, we first use analytic methods to generate viable grasps for each single object, and reject unfeasible grasps in densely clutter scenes. To the best of our knowledge, the dataset we generated is the first large-scale synthetic 6-DoF grasp dataset for dense clutters.

Deep Learning on 3D Data Qi et al. [21, 22] proposed PointNet and PointNet++, a novel 3D deep learning network architecture capable of extracting useful representations from 3D point clouds. Compared with other architectures [23, 24], PointNets are robust to varying sampling densities, which is important to real robotic applications. In this paper, we utilize PointNet++ as the backbone of our single-shot grasp detection and demonstrate its effectiveness.

3 Problem Setting

We denote the single-view point cloud by $\mathbf{P}$ and the gripper description by $\mathbf{G}$. A parallel gripper can be parameterized by the frame whose origin lies at the middle of the line segment connecting two figure tips and orientation aligns with the gripper axes. We therefore denote a grasp configuration as $c = (\mathbf{h}, s_{\mathbf{h}})$, where $\mathbf{h} \in \mathbb{SE}(3)$ and $s_{\mathbf{h}} \in \mathbb{R}$ is a score measuring the quality of $\mathbf{h}$.

4 Training Data Generation

To train our S^{4}G, a large scale dataset capturing cluttered scenes, with viable grasps and quality scores as groundtruth, is indispensable. Fig. 2 illustrates the training data generation pipeline. We use the YCB object dataset [25] for our data generation. Since S^{4}G directly takes a single-view point cloud from the depth sensor as input and outputs collision-free grasps in a densely-cluttered environment, we need to generate such scenario with complete scene point cloud and corresponding partially observed point cloud. Each point in the point cloud is assigned with serval grasps which will be introduced in Sec 4.3 and each ground truth grasp has a $\mathbb{SE}(3)$ pose, an antipodal score, a collision score, an occupancy score, and a robustness score, which we will introduce later. On the other hand, the scene point cloud does not interact with the network explicitly, but it serves as a reference to evaluate grasps in the point cloud.

4.1 Gripper Contact Model

Vast literature exists to find regions suitable to grasp by analyzing the 3D geometry [26]. Among these methods, force closure has been widely used to synthesize grasps and can be reduced into calculating angles between face normals, known as antipodal grasp [27, 28]. Here we introduce our gripper contact model based on force closure analysis to find feasible grasps.

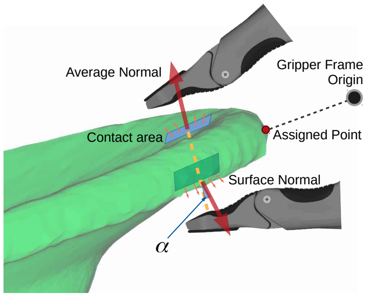


Figure 3: Illustration of Gripper Contact Model

To be more specific, we first detect all possible contact pairs with high antipodal score $s^a_{\mathbf{h}} = \mathbf{cos}(\alpha_1)\mathbf{cos}(\alpha_2)$, where α_i is the angle between the outward normal and the line connecting two contact points. As illustrated in Fig. 3, for each contact pair (p_i, p_j), the normal $\mathbf{n}_i$ at point p_i is smoothed with radius $r$mm. Note that this step is important to grasp objects of rugged surface with high-frequency normal variation. However, we do not directly use a ball query to query its neighbors, which will lead to undesirable results at corners and edges. Instead, we remove the neighbors which has a distance along the normal direction (calculated as $r^k_i = |(\mathbf{p}^k_i - \mathbf{p}_i) \cdot \frac{\mathbf{n}_i}{|\mathbf{n}_i|}|$) larger than 3mm in the query ball of radius r for normal calculation, where $\mathbf{p}^k_i$ is the k-th neighbor of point i.

These two hyper-parameters have definite physical meaning, which is distinct from the approach to obtain the gripper contact model hyper-parameters in GPD [10] through extensive parameter tuning. As shown in Fig. 3, our gripper will only interact with the object by its soft rubber pad, which allows deformation within 3mm. And the normal smoothing radius is set as the gripper width $r = 23$mm.

In fact, our gripper model has clear advantage over Darboux frame based methods, especially at rugged surfaces and flat surfaces. For rugged surfaces, there is no principled way to decide the radius for normal smoothing, since the radius is not only relevant to the gripper, but also to the object to grasp. For flat surfaces, the principal curvature directions are under-determined. In practice, we do observe issues for these cases. For example, for plates and mugs, Darboux frame based method will likely to fail in generating a successful grasp pose for the thin wall.

Besides the direction of contact force, we also consider the stability of the grasp. The occupancy score $s^o_{\mathbf{h}}$, which represents the volume of object within the gripper closing region $\mathbf{R}(\mathbf{c})$, is calculated by

$$s^o_{\mathbf{h}} = \min\{ln(|\mathbf{P_{close}}|), 6\}, \quad \mathbf{P_{close}} = \mathbf{R}(\mathbf{c}) \cap \mathbf{P}, \tag{1}$$

where $\mathbf{P_{close}}$ is the number of points within closing region. If $s^o_{\mathbf{h}}$ is small, the gripper contact analysis will be unreliable. To make sure that the point cloud occupancy can correctly represent the volume, we down-sample the point cloud using voxel grid filter with a leaf size of 5mm.

4.2 Physically-plausible Scene Synthesis from Objects

Since our network is trained on synthesis data and directly applied to real world scenarios, it is necessary to generate training data closer to reality both physically and visually.

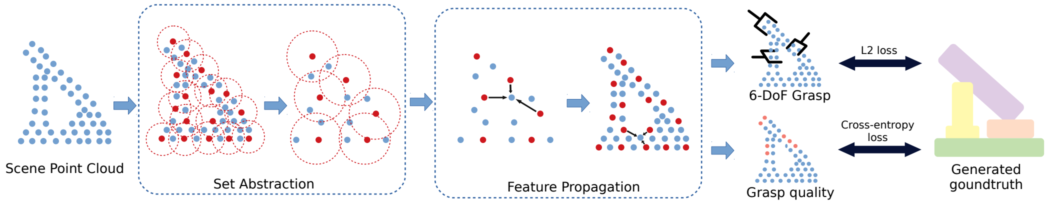


Figure 4: Architecture of Single-Shot Grasp Proposal Network based on PointNet++ [22]. Given the scene point cloud, our network first extracts hierarchical point set features by progressively encoding points in larger local regions; then the network propagates the point set features to all the original points using inverse distance interpolation and skip links; finally it predicts one 6-DoF grasp pose $\mathbf{h}_i$, and one grasp quality score $s_{\mathbf{h}_i}$ of every point.

We need physically-plausible layouts of various scenes where each object should be in equilibrium under gravity and contact force. Therefore, we adopt MuJoCo engine [29] and V-HACD[30] to generate scenes where each object is in equilibrium. Objects initialized with random elevation and poses fall onto a table in the simulator and converge to static equilibrium due to friction. We record the poses and positions of objects and reconstruct the 3D scene.(Fig. 2)

Beside scene point cloud, we also need to generate viewed point clouds that will feed into the neural network. To simulate the noise of depth sensor, we apply a noise model on the distance from camera optical center to each point as $\tilde{D}_{o,p} = (1 + \mathcal{N}(0, \sigma^2))D_{o,p}$, where $D_{o,p}$ is the noiseless distance captured by a ray tracer and $\tilde{D}_{o,p}$ is the distance used to generate viewed point clouds. We employ $\sigma = 0.003$ in this paper.

4.3 Robustness Grasp Generation by Scene Analysis

Given the scene point cloud, we can do collision detection for each grasp configurations. Collision score $s^c_{\mathbf{h}}$ is a scene-specific boolean mask indicating the occurrence of collision between the proposed gripper pose and the complete scene. As shown in our experiment, our network can better predict collision with invisible parts.

It is a common case that robot end-effector can not move precisely to a given pose due to sensor noise, hand-eye calibration error and mechanical-transmission noise. To perform a successful grasp under imperfect condition, the proposal grasp should be robust enough against gripper's pose uncertainty. In this paper, we add a small perturbation to the $\mathbb{SE}(3)$ grasp pose and evaluate the antipodal score, occupancy score and collision score for the perturbed pose. The final scalar score of each grasp can be derived as:

$$s_{\mathbf{h}} = \min_j [s^a_{\mathbf{h}_j} s^o_{\mathbf{h}_j} s^c_{\mathbf{h}_j}], \quad \mathbf{h}_j = \mathbf{exp}(\hat{\xi})\mathbf{h}, \tag{2}$$

where $\hat{\xi} \in \mathfrak{se}(3)$ is the pose perturbation and $\mathbf{exp}$ is the exponential mapping. The final viewed point cloud with ground truth grasps and scores will serve as training data for our S^{4}G.

5 Single-Shot Grasp Generation

5.1 PointNet++ based Grasp Proposal

We design the single-shot grasp proposal network based on the segmentation version of PointNet++, which has demonstrated state-of-the-art accuracy and strong robustness over clutter, corruption, non-uniform point density [22], and adversarial attacks [31].

Figure. 4 demonstrates the architecture of S^{4}G, which takes the single-view point cloud as input, and assigns each point two attributes. The first attribute is a good grasp (if exists) associated to the point by inverse indexing, and the second attribute is the quality score of the stored grasp. The generation of the grasp and quality score can be found in Sec. 4.3.

The hierarchical architecture not only allows us to extract local features and predict reasonable local frames when the observation is partial and noisy, but also combines local and global features to effectively infer the geometry relationship between objects in the scene.

Compared with sampling and grasp classification [10, 18], the single-shot 6-DoF grasp direct regression task is more challenging for networks to learn, because widely adopted rotation representations such as quaternions and Euler angles are discontinuous. In this paper, we use a 6D representation of the 3D rotation matrix because of its continuity [32]: for every $\mathbf{R} \in \mathbb{SO}(3)$, it is represented by $\mathbf{a} = [\mathbf{a}_1, \mathbf{a}_2]$, $\mathbf{a}_1, \mathbf{a}_2 \in \mathbb{R}^3$, such that the mapping $f : \mathbf{a} \to \mathbf{R}$ is

$$\begin{aligned} &\mathbf{R} = [\mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3] \\ &\mathbf{b}_1 = N(\mathbf{a}_1) \\ &\mathbf{b}_2 = N(\mathbf{a} - \langle \mathbf{a}_2, \mathbf{b}_1 \rangle \mathbf{b}_1) \\ &\mathbf{b}_3 = \mathbf{b}_1 \times \mathbf{b}_2, \end{aligned} \tag{3}$$

where $N()$ denotes the normalization function. Because the gripper is symmetric with respect to rotation around the x axis, we use a loss function which handles the ambiguity by considering both correct rotation matrices as ground truth options. Given the groundtruth rotation matrix $\mathbf{R}_{GT}$, we define the rotation loss function L_{rot} as

$$\begin{aligned} L_{rot} &= \min_{i \in \{0,1\}} \| f(\mathbf{a}_{pred}) - \mathbf{R}_{GT}^{(i)} \|^2 \\ \mathbf{R}_{GT}^{(i)} &= \mathbf{R}_{GT} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\pi i) & 0 \\ 0 & 0 & \cos(\pi i) \end{bmatrix} \end{aligned} \tag{4}$$

The prediction of translation vectors is treated as a regression task and the L_2 loss is applied. By dividing the groundtruth score into multiple levels, the grasp quality score prediction is treated as a multi-class classification task, and a weighted cross-entropy loss is applied to handle the unbalance between positive and negative data. We only supervise the pose prediction for those points assigned with viable grasps and the total loss is defined as:

$$L = \sum_{\mathbf{P}_v} (\lambda_{rot} \cdot L_{rot} + \lambda_t \cdot L_t) + \sum_{\mathbf{P}_s} (\lambda_s \cdot L_s), \tag{5}$$

where $\mathbf{P}_v, \mathbf{P}_s$ represent the point set with viable grasps and the whole scene point cloud, respectively. $\lambda_{rot}, \lambda_t, \lambda_s$ are set to 5.0, 20.0, 1.0 in experiments.

5.2 Non-maximum Suppression and Grasp Sampling

Algorithm. 1 describes the strategy to choose one grasp execution **h** from the network prediction $\mathcal{C}$.

Algorithm 1: NMS and Grasp sampling

```
Input: Prediction $\mathcal{C}$: $\{(\mathbf{h}_i, s_{\mathbf{h}_i})\}$
Export: Grasp Execution: $h$
  Executable Grasps $\mathcal{H} = \{\}$
  $Sort\ \{(\mathbf{h}_i, s_{\mathbf{h}_i}\}\ by\ s_{\mathbf{h}_i}$
  $i = 0$
  while $\mathcal{L}ength(\mathcal{H}) < N$ do
    if $(Collision == False)$ and
    $\underset{min}{\mathbf{h}_k \in \mathcal{H}} dist(\mathbf{h}_i, \mathbf{h}_k) > \epsilon$ then
      $Add\ (\mathbf{h}_i, s_{\mathbf{h}_i})$ to $\mathcal{H}$
    end if
    $i = i + 1$
  end while
  $p_k = \frac{g(s_{\mathbf{h}_k})}{\sum_l g(s_{\mathbf{h}_l})}$ for $\mathbf{h}_k \in \mathcal{H}$
  while Motion planning fails do
    Sample $h$ according to $\{p_k\}$
  end while
```

Because the network generates one grasp for each point, there are numerous similar grasps in each grasp's neighborhood and we use non-maximum suppression (NMS) to select grasps **h** with local maximum $s_{\mathbf{h}_i}$ to generate executable grasp set $\mathcal{H}$. Then weighted random sampling is applied to sample one grasp to execute according to its grasp quality score.

6 Experiments

6.1 Implementation Details

The input point cloud is first preprocessed, including workspace filtering, outliers removal, and voxel grid down-sampling. For training and validation, we sample $\frac{1}{8}N$ points from the point set with viable grasps, $\frac{7}{8}N$ from the remaining point set, and integrate them as the input of the network. For evaluation, we sample N points at random from the preprocessed point cloud. N is set to 25600 in our experiments. We implement our network in PyTorch, and train it using Adam [33] as the optimizer for 100 epochs with the initial learning rate 0.001, which is decreased by 2 every 20 epochs.

6.2 Superiority of $\mathbb{SE}(3)$ grasp

We first evaluated the grasp quality performance of our proposed network on simulated data. To demonstrate the superiority of $\mathbb{SE}(3)$ grasp over 3/4 DoF grasp, here we give a quantitative analysis over 6k scene with around 2.6M generated grasps (Fig. 5). In our experiments, grasps are uniformly divided into 6 groups according to the angle between the approach vector and vertical direction in the range of $(0°, 90°)$. We use the recall rate as metric which are defined as the percentage of objects that can be grasped using grasps between vertical and certain angle. We evaluate the recall rate at scenes of three different densities: simple (1-5 objects presented in the scene), semi-dense (6-10 objects) and dense (11-15) objects. The overall recall rate is the weighted average of the three scenes. We find that only 63.38% objects can be grasped by nearly vertical grasps $(0°, 15°)$. With the increase of scene complexity, the advantage of $\mathbb{SE}(3)$ grasp becomes more remarkable.

6.3 Simulation Experiments

GPD [10] and PointNetGPD [18] adopt Darboux frame analysis to sample grasp poses and train a classifier to evaluate their quality, which achieved state-of-the-art performance in 6D grasp detection. We choose GPD(3 channels), GPD(12 channels), and PointGPD as our baseline methods. For training baseline methods, we adopt their grasp sampling strategy to generate grasp candidates for each scene until we get 300 collision-free grasps. We generate grasps over 6.5k scenes and get more than 2M grasps, which is larger than the 300K grasps in the original paper. Note that the scene used to generate training data for baseline method is exactly the same as our method.

For evaluation of baseline methods, we first sample 1000 points at random from the point cloud and calculate the Darboux frame for grasp candidates, which are then classified and ranked. The top 10 grasps are evaluated for both baseline methods and our methods.

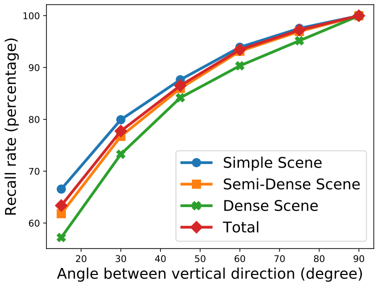


Figure 5: Results of simulated experiment on the recall rate of different grasp. The X-axis is the angle between the approach vector and vertical direction. The angle of absolute 3/4 DoF grasp is 0°

To evaluate our method in finding collision-free grasps, we compare two metrics that affect the final grasping success rate: **(1)** antipodal score, which describes the force closure property of grasps, **(2)** probability of collision with other objects not observable to the depth sensor. The evaluation is performed in simulator with 2 settings: **(1)** No noise, where the point cloud from the depth sensor simulator aligns with the complete point cloud perfectly; **(2)** With noise, where the noise of the depth simulator is proportional to the depth. Please note that for noise setting and real-world experiment, both baselines and our method is trained on noisy data. Table. 1 shows the comparison results. Since the 6-DoF grasp pose is regressed by our S^{4}G instead of being computed from local normals and curvatures, it is less sensitive to partial and noisy depth observations; also, our S^{4}G is able to generate more collision-free grasps by inferring from local and global geometry information jointly.

6.4 Robotic Experiments

We validate the effectiveness and reliability of our methods in real robotic experiments. We carried out all the experiments on Kinova MOVO, a mobile manipulator with a Jaco2 arm attached with a 2-finger gripper (Fig. 1 (a)). In order to be close to real domestic robot application scenarios, we

	w/o Noise		**w/ Noise**	
	Antipodal Score	Collision-free	Antipodal Score	Collision-free
GPD (3 channels)	0.5947	**47.07%**	0.5802	40.00%
GPD (12 channels)	0.5883	45.27%	0.5946	40.44%
PointNetGPD	0.5718	42.41%	0.6376	41.17%
Ours	**0.7364**	47.02%	**0.7354**	**53.32%**

Table 1: Comparison of grasp quality on simulation data.

	Grasp quality		**Time-efficiency**		
	Success rate	Completion rate	Processing	Inference	**Total**
GPD (3 channels)	40.0%	60.0%	24106 ms	**1.50 ms**	24108 ms
GPD (12 channels)	33.3%	50.0%	27195ms	1.70ms	27197ms
PointNetGPD	40.0%	60.0%	17694ms	2.86ms	17697ms
Ours	**77.1%**	**92.5%**	**5804ms**	12.60 ms	**5817 ms**

Table 2: Results of robotic experiments on dense clutters. *Success rate* and *completion rate* are used as the evaluation metrics, which represent the accuracy and completeness respectively.

use one KinectV2 depth sensor that is mounted on the head of the manipulator, which makes the observation heavily occluded and raises the difficulty of experiments. 30 objects of various shapes and weight (see Supplymentary Materials) are used, which are absent in the training dataset.

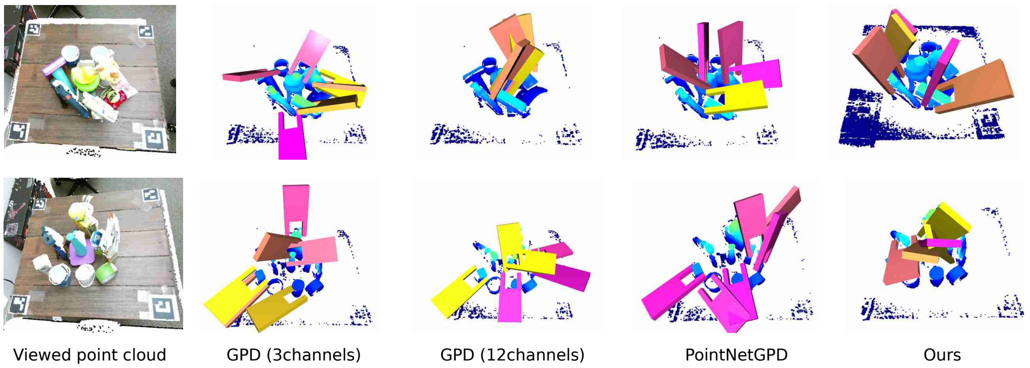
The experiment procedure is as follows: **(1)** Choose 10 out of the 30 objects at random and put them on the table to form a cluttered scene; **(2)** The robot attempts multiple grasps, until all objects are grasped or 15 grasps have been attempted; **(3)** Step (1) and (2) are repeated for 4 times for each method. More details are presented in the supplementary material. Note that all the objects selected in real robot experiments are out of the training data.

As illustrated in Table 2, our method outperforms baseline methods in terms of *success rate*, *completion rate*, and time efficiency, which suggests that the single-shot regressed 6-DoF grasps have better force closure quality than sampled grasps from baselines, as demonstrated in Fig. 6. Not needed by us, the baseline methods also need to detect collision and extract local geometry for every sampled grasp, which takes around 20 seconds, so they are much more time-consuming than our method.

Our experiment setting is much more challenging than the baseline papers. In the original paper, GPD uses two depth sensors at both sides of the arena to capture the nearly complete point cloud in the original paper, but in our experiments, only one depth sensor is used. In both baselines, grasps are sampled in the neighbourhood of Darboux frame. It performs well on convex objects (box and ball) but poorly on non-convex or thin-structure objects, such as mug and bowl as in our experiments, because their heuristic sampling method requires accurate normals and curvatures but estimating those surface normals from noisy point cloud is challenging. On the contrary, PointNet++ has been demonstrated to be robust against adversarial changes to the input data [31], which can better capture the geometric structure under noise.

7 Conclusion

We studied the problem of 6-DoF grasping by a parallel gripper in a cluttered scene captured using a commodity depth sensor from a single viewpoint. Our learning based approach trained in a synthetic scene can work well in real-world scenarios, with improved speed and success rate compared with state-of-the-arts. The success shows that our design choices, including a single-shot grasp proposal and a novel gripper contact model, are effective.



Viewed point cloud

GPD (3channels)

GPD (12channels)

PointNetGPD

Ours

Figure 6: Comparison between sampled grasps chosen by baseline methods with high-score and regressed grasps by our method.

Acknowledgments

We would like to acknowledge the National Science Founding for the grant RI-1764078 and Qualcomm for the generous support. We especially thank Jiayuan Gu for the discussion on network architecture design and Fanbo Xiang for the idea on using single object cache to accelerate training data generation.

References

[1] J. Bohg, A. Morales, T. Asfour, and D. Kragic. Data-driven grasp synthesisa survey. *IEEE Transactions on Robotics*, 30(2):289–309, 2013.

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

[3] H. Dang and P. K. Allen. Semantic grasping: Planning robotic grasps functionally suitable for an object manipulation task. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1311–1317. IEEE, 2012.

[4] 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*, 37(4-5):421–436, 2018.

[5] E. Jang, C. Devin, V. Vanhoucke, and S. Levine. Grasp2vec: Learning object representations from self-supervised grasping. *arXiv preprint arXiv:1811.06964*, 2018.

[6] D. Watkins-Valls, J. Varley, and P. Allen. Multi-modal geometric learning for grasping and manipulation. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 7339–7345. IEEE, 2019.

[7] D. Quillen, E. Jang, O. Nachum, C. Finn, J. Ibarz, and S. Levine. Deep reinforcement learning for vision-based robotic grasping: A simulated comparative evaluation of off-policy methods. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6284–6291. IEEE, 2018.

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

[9] S. Kumra and C. Kanan. Robotic grasp detection using deep convolutional neural networks. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 769–776. IEEE, 2017.

[10] A. ten Pas, M. Gualtieri, K. Saenko, and R. Platt. Grasp pose detection in point clouds. *The International Journal of Robotics Research*, 36(13-14):1455–1473, 2017.

[11] M. Gualtieri and R. Platt. Learning 6-dof grasping and pick-place using attention focus. *arXiv preprint arXiv:1806.06134*, 2018.

[12] J. Mahler, M. Matl, V. Satish, M. Danielczuk, B. DeRose, S. McKinley, and K. Goldberg. Learning ambidextrous robot grasping policies. *Science Robotics*, 4(26):eaau4984, 2019.

[13] E. Johns, S. Leutenegger, and A. J. Davison. Deep learning a grasp function for grasping under gripper pose uncertainty. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4461–4468. IEEE, 2016.

[14] S. Caldera, A. Rassau, and D. Chai. Review of deep learning methods in robotic grasp detection. *Multimodal Technologies and Interaction*, 2(3):57, 2018.

[15] A. Collet, M. Martinez, and S. S. Srinivasa. The moped framework: Object recognition and pose estimation for manipulation. *The International Journal of Robotics Research*, 30(10): 1284–1306, 2011.

[16] A. Zeng, K.-T. Yu, S. Song, D. Suo, E. Walker, A. Rodriguez, and J. Xiao. Multi-view self-supervised deep learning for 6d pose estimation in the amazon picking challenge. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1386–1383. IEEE, 2017.

[17] A. Mousavian, C. Eppner, and D. Fox. 6-dof graspnet: Variational grasp generation for object manipulation. *arXiv preprint arXiv:1905.10520*, 2019.

[18] H. Liang, X. Ma, S. Li, M. Görner, S. Tang, B. Fang, F. Sun, and J. Zhang. PointNetGPD: Detecting grasp configurations from point sets. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2019.

[19] F.-J. Chu, R. Xu, and P. A. Vela. Real-world multiobject, multigrasp detection. *IEEE Robotics and Automation Letters*, 3(4):3355–3362, 2018.

[20] A. Bicchi and V. Kumar. Robotic grasping and contact: A review. In *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065)*, volume 1, pages 348–353. IEEE, 2000.

[21] C. R. Qi, H. Su, K. Mo, and L. J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. *arXiv preprint arXiv:1612.00593*, 2016.

[22] C. R. Qi, L. Yi, H. Su, and L. J. Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *Advances in Neural Information Processing Systems*, pages 5099–5108, 2017.

[23] 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)*, pages 922–928. IEEE, 2015.

[24] C. R. Qi, H. Su, M. Nießner, A. Dai, M. Yan, and L. J. Guibas. Volumetric and multi-view cnns for object classification on 3d data. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5648–5656, 2016.

[25] 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)*, pages 510–517. IEEE, 2015.

[26] A. Sahbani, S. El-Khoury, and P. Bidaud. An overview of 3d object grasp synthesis algorithms. *Robotics and Autonomous Systems*, 60(3):326–336, 2012.

[27] V.-D. Nguyen. Constructing force-closure grasps. *The International Journal of Robotics Research*, 7(3):3–16, 1988.

[28] I.-M. Chen and J. W. Burdick. Finding antipodal point grasps on irregularly shaped objects. *IEEE transactions on Robotics and Automation*, 9(4):507–512, 1993.

[29] E. Todorov, T. Erez, and Y. Tassa. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5026–5033. IEEE, 2012.

[30] K. Mamou and F. Ghorbel. A simple and efficient approach for 3d mesh approximate convex decomposition. In *2009 16th IEEE international conference on image processing (ICIP)*, pages 3501–3504. IEEE, 2009.

[31] D. Liu, R. Yu, and H. Su. Extending adversarial attacks and defenses to deep 3d point cloud classifiers. *arXiv preprint arXiv:1901.03006*, 2019.

[32] Y. Zhou, C. Barnes, J. Lu, J. Yang, and H. Li. On the continuity of rotation representations in neural networks. *arXiv preprint arXiv:1812.07035*, 2018.

[33] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

A Supplementary Material

A.1 Network Details

We use 3 point set abstract layers, each of which is a 3-layer MLP, containing $(128, 128, 256)$, $(256, 256, 512)$, $(512, 512, 1024)$ units, respectively. ReLU is used as the activation function. Farthest Point Sampling(FPS) is adopted for better and more uniform coverage, where a subset of points are chosen from the input point set such that each point in the subset is the most distant point from points in the set. Compared with random sampling, FPS has better coverage of the entire point set. It is performed iteratively to get the centroids for grouping from the former stage.

A.2 Robotics Experiments Dataset

Figure. 7 shows the 30 objects used in our experiments. This dataset is collected from daily objects and different from the YCB[25] dataset we used to generate training data.

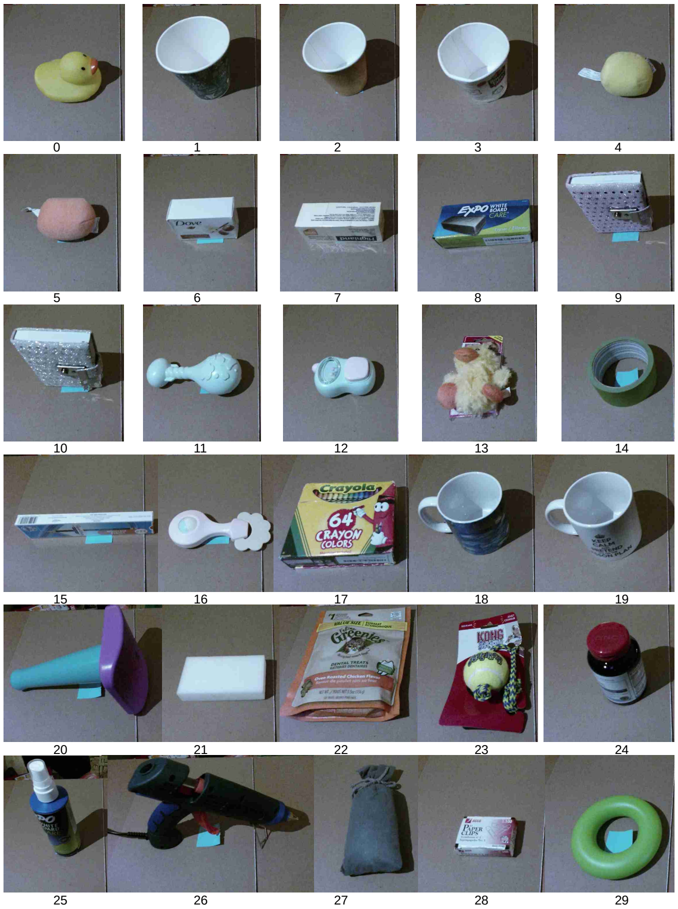


Figure 7: The 30 objects used in our experiments.

A.3 Robotics Experiments Grasp Proposal

Figure. 8, 9 show the viewed point cloud and proposed high quality grasp set in robotic experiments.

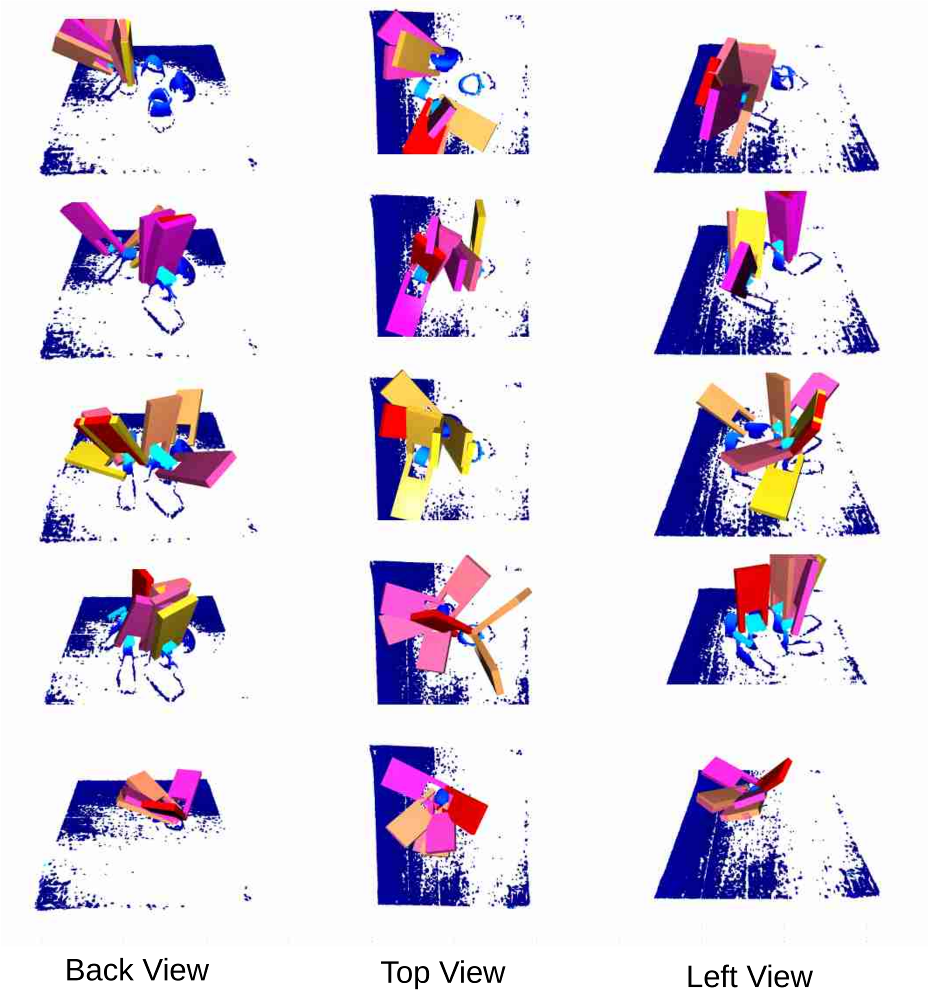


Figure 8: Viewed point cloud from the depth sensor and high quality grasp set in robotic experiments.

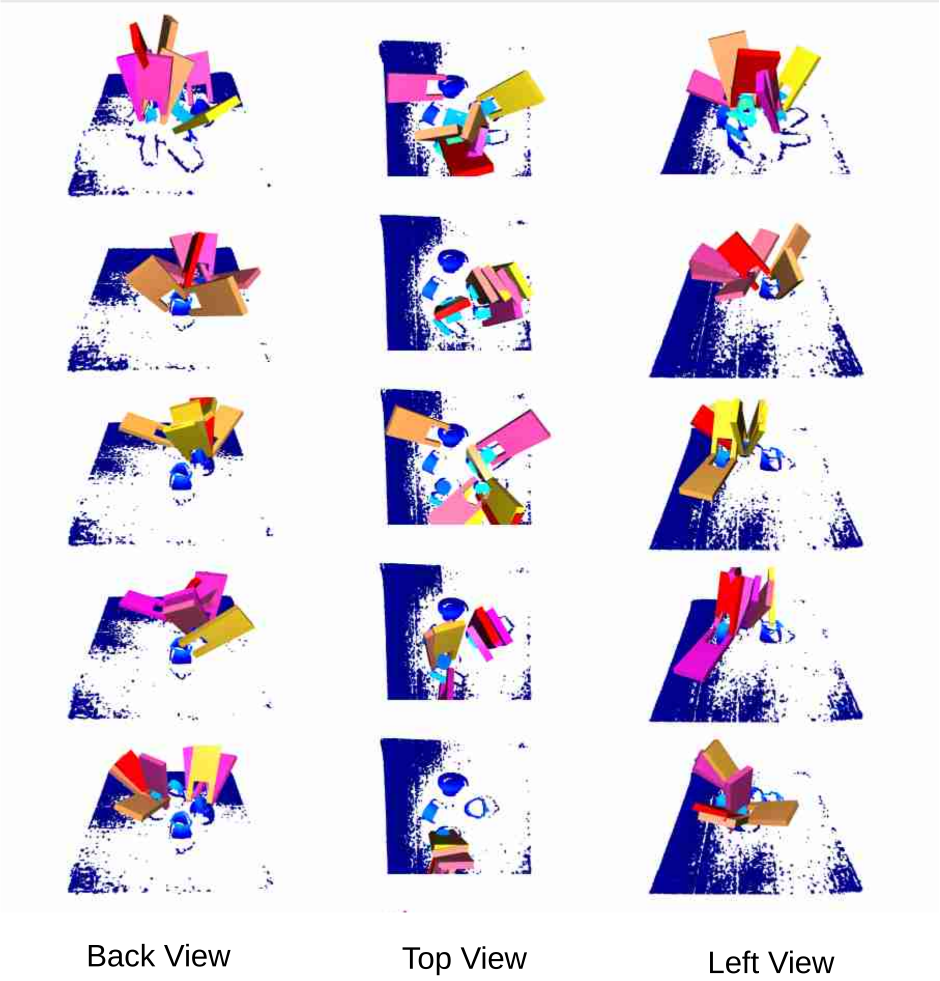


Figure 9: More viewed point cloud from the depth sensor and high quality grasp set in robotic experiments.