IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2023, London, UK. Cite as RA-L paper.

SymmetryGrasp: Symmetry-aware Antipodal Grasp Detection from Single-view RGB-D Images

Yifei Shi, Zixin Tang, Xiangting Cai, Hongjia Zhang, Dewen Hu and Xin Xu

***Abstract*—Symmetry is ubiquitous in everyday objects. Humans tend to grasp objects by recognizing the symmetric regions. In this paper, we investigate how symmetry could boost robotic grasp detection. To this end, we present a learning-based method for detecting grasp from single-view RGB-D images. The key insight is to explicitly incorporate symmetry estimation into grasp detection, improving the quality of the detected grasps. Specifically, we first introduce a new grasp parameterization in grasp detection for parallel grippers based on symmetry. Based on this representation, a symmetry-aware grasp detection network method is present to simultaneously estimate object symmetry and detect grasp. We find that the learning of grasp detection greatly benefits from symmetry estimation, improving the training efficiency and the grasp quality. Besides, to facilitate the cross-instance generality of grasping unseen objects, we propose Principal-directional scale-Invariant Feature Transformer (PIFT), a plug-and-play module, that allows spatial deformation of points during the feature aggregation. The module essentially learns feature invariance to anisotropic scaling along the shape principal directions. Extensive experiments demonstrate the effectiveness of the proposed method. In particular, it outperforms previous methods, achieving state-of-the-art performance in terms of grasp quality on GraspNet-1-Billion and success rate on a real robot grasping experiment.**

***Index Terms*—RGB-D Perception, Deep Learning in Grasping and Manipulation, Deep Learning for Visual Perception**

I. Introduction

As one of the most fundamental skills for intelligent robots, grasping has a wide range of applications from bin-picking for industrial robots to general object grasping for home service robots. With the recent progress of RGB-D sensing and 3D learning techniques, detecting feasible grasp from cluttered scenes has attracted a surge of research attention from robotics and computer vision communities.

Recent advances in grasp detection are dominated by learning-based approaches. According to whether the object geometry is known beforehand or not, these approaches can be divided into two categories: the model-based methods [36] and the model-free methods [9]. While the former is not always applicable to novel object grasping, the latter requires a large number of data with careful annotations to be trained with. In particular, those methods are prone to fall short in grasping novel objects located in cluttered scenes, due to the inferior generality caused by insufficient network training. To solve this problem, recent research has been focused on improving the quality of the train data [38], [9] or involving the advanced learning models [18], [24].

Manuscript received: June, 1, 2022; Revised August, 25, 2022; Accepted September, 26, 2022.

This paper was recommended for publication by Editor Cesar Cadena Lerma upon evaluation of the Associate Editor and Reviewers' comments. This work was supported in part by NSFC (61825305, 62002379) and National Key Research and Development Program of China (2018YFB1305105).

The authors are with the College of Intelligence Science and Technology, National University of Defense Technology, China.

Corresponding author: Xin Xu (xinxu@nudt.edu.cn).

Digital Object Identifier (DOI): see top of this page.

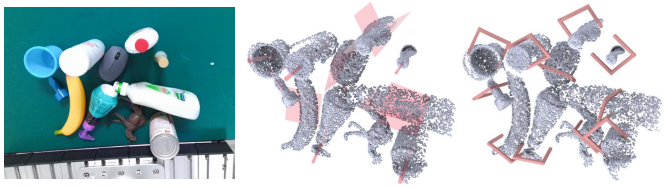

Fig. 1: SymmetryGrasp couples the detection of symmetry and grasp into a unified framework. Given a single-view RGB-D image (left), it takes advantage of the estimated symmetries (middle) to determine the feasible antipodal grasps (right), greatly increasing grasp quality and generality.

In this paper, we study the problem of grasp detection from a new perspective. The key insight is simple: *symmetry is prevalent in everyday objects, and humans tend to grasp objects with fingers placed at symmetric regions*. Theoretically, symmetry detection and grasp detection are highly relevant tasks in both psychology [16], [8] and robotics [1], [2]. Besides, from the prospect of geometry learning, symmetry is a lightweight yet informative representation of 3D geometry, which can potentially narrow the searching space of grasp detection. As such, understanding object symmetry seems to be an intermediate and complementary stage of grasp detection. It is therefore interesting to study whether symmetry detection could be incorporated into the grasp detection pipelines and boost the performance.

We present SymmetryGrasp, a learning-based method for detecting antipodal grasp from single-view RGB-D images with the guidance of symmetry estimation (Figure 1). Our contributions are two-fold. First, we introduce a new parameterization of grasp detection for parallel grippers based on symmetry. The parameterization is designed to explicitly exploit symmetry clues and is expected to improve the robustness and generality of general grasp detection. Based on this parameterization, a symmetry-aware grasp detection network method is presented to simultaneously estimate symmetry and detect grasp. The detected grasps are located in local symmetric regions and regularized by the estimated symmetry. With a dedicated network design, the network is able to detect plausible grasps on objects with multiple symmetry types, i.e. reflection, rotation, and sphere. Since the network is even applicable to objects whose symmetric region only occupies a small portion, it is not only designed for grasping symmetric objects but provides a universal solution for general grasp detection.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2023, London, UK. Cite as RA-L paper.

Second, to improve the cross-instance generalization ability in grasping the unseen objects, we propose *Principal-directional scale-Invariant Feature Transformer (PIFT)*, a plug-and-play module for point convolutional networks. PIFT allows spatial deformation of points in the point convolution processing, facilitating feature invariance to scaling along principal directions of the symmetry plane/axis. Equipped with PIFT, SymmetryGrasp is able to detect high-quality grasps on novel objects.

Extensive experiments demonstrate that SymmetryGrasp is able to estimate symmetry and detect grasp in various scenes with good performance. In particular, the learning of grasp detection greatly benefits from symmetry estimation. SymmetryGrasp is robust in detecting grasps on objects with heavy occlusion and novel geometry, outperforming the prior methods. It achieves state-of-the-art performance in terms of grasp quality on GraspNet-1-Billion and success rate on a real robot grasping experiment.

II. Related Work

A. Learning-based Grasp Detection

Learning-based grasp detection methods can be roughly classified into two categories: model-based and model-free. Model-based methods [3], [41], [36], [31] rely on the pre-scanning of the target objects. The plausible grasps are generated by the mechanics-based approaches and are transferred into the captured data, usually by feature matching techniques, during inference. The drawback of model-based grasp detection methods is that they can hardly generalize to unseen objects whose 3D model is inaccessible. Conversely, model-free methods do not require the 3D model of the target objects to be fully known and are thus more general. The early research in this direction focuses on detecting planar grasp from single-view or multi-view images [17], [29], [21], [20], [14]. Recently, several algorithms and datasets have been proposed to learn high-DoF grasps [35], [25], [28], [11], [43]. Our method falls into this category. However, It is different to the previous works since it explicitly incorporates symmetry estimation into the grasp detection workflow, resulting in grasp detection with higher quality.

B. Symmetry Detection

Symmetry is a universal concept and is deemed to be one of the most basic geometric clues for object understanding [23], [30]. As such, symmetry detection could potentially benefit many downstream applications [26], such as shape completion [34] and pose estimation [5]. Prior works of symmetry detection are only applicable to synthetic objects in the virtual world, limiting the effectiveness of robotic applications. Several recent works studies detecting symmetry from partially observed data with various formats, such as RGB image [42], RGB-D image [32], [33], and 3D volume [10]. Our method is inspired by the above methods. However, it requires a significantly smaller number of training data, thanks to the designs on feature aggregation.

III. Method

A. Problem Setting

Our goal is to grasp objects with a robot arm. The robot arm is equipped with a parallel gripper. A depth camera is deployed around the robot arm to acquire RGB-D images. We assume the geometry of the gripper, the intrinsics of the camera, as well as the transformation between the camera and the robot arm, are already known. The objects are cluttered with each other, incurring mutual object occlusions and various object poses.

Our method detects grasps from a single-view RGB-D image. To be specific, it generates a 7D grasp (R, t, d) from each of the objects from the acquired RGB-D image I, where R, t and d are the 3D rotation, the 3D translation, and the opening width of the two-finger gripper, respectively. R and t together determine a rigid transformation from the model coordinate of the gripper to the camera coordinate.

B. Symmetry-guided Grasp Detection Parameterization

Directly estimating all 7 DoFs of the grasp is hard. We propose a decomposed grasp parameterization in grasp detection guided by symmetry. The idea is inspired by human behaviors on grasp. In order to perform robust grasping, humans are likely to first infer the contact points on the objects with the fingers (i.e. where to grasp), and then determine the direction of our hand to approach these contact points (i.e. how to grasp). We mimic the procedure and decompose the estimation of the 7D grasp into two stages.

On the first stage, the locations of the two contact points are estimated, determining 6 DoFs (i.e. 3 DoFs in translation, 2 DoFs in rotation, and 1 DoF in opening width):

$$\alpha, \beta, t, d = F(I), \tag{1}$$

where α and β are the two Euler angles of the rotation R. In particular, we select one of the contact points from the input points and estimate its symmetric counterpart to be another one. The translation t and opening width d could then be determined accordingly.

On the second stage, the rotation of gripper around the line that connects the two contact points is estimated (i.e. 1 DoF in rotation):

$$\gamma = G(I, \alpha, \beta, t, d), \tag{2}$$

where γ is the third Euler angle of rotation, determining the in-plane rotation.

The symmetry-guided grasp parameterization has the following advantages. *First*, it has a clear and natural task division that makes the first stage focus on intra-object geometry and the second stage analyze the inter-object layout, facilitating a more effective network training. *Second*, since it explicitly takes the symmetry as a prior, the grasp detection is well-constraint and therefore expected to be more accurate and stable. *Third*, it allows the gripper to contact with the symmetric counterparts which might be located in the unseen region, the grasping location is not restricted to the observed points, expanding the manipulation range of the robot.

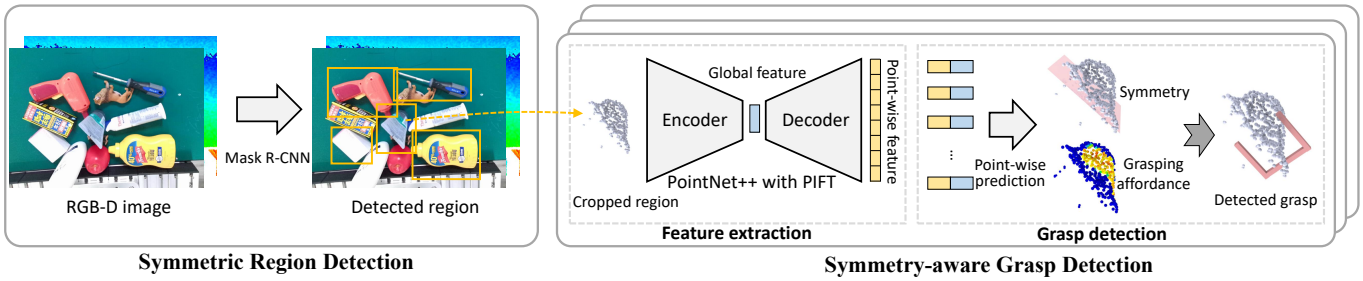

Fig. 2: Method overview. Our method first detects symmetric regions from the RGB-D image with a Mask R-CNN. Each detected region is then transferred into the point cloud and fed into a symmetry-aware grasp detection network. The symmetry-aware grasp detection network extracts features by a point convolutional network with PIFT modules that allow non-rigid shape deformation during feature aggregation. It outputs both the symmetry and grasp affordance, which together correspond to the detected grasp.

C. Method Overview

We present SymmetryGrasp to implement the above idea. The method starts by performing an instance segmentation to detect symmetric regions in the scene. Then, a symmetry-aware grasp detection network (i.e. $F(\cdot)$) is proposed to detect the contact points of the gripper. That is achieved by detecting symmetry and estimating the point-wise grasp affordance with a unified network. Last, a physics-based grasp selection algorithm (i.e. $G(\cdot)$) is present to generate plausible grasps. The overview of the method is illustrated in Figure 2. In the following, we describe each step in detail.

D. Symmetric Region Detection

The first step is to detect graspable objects or regions in cluttered scenes. Existing works on robotic grasping have demonstrated that it is possible to detect grasps without performing an object detection [39], [40]. Nevertheless, these methods need to be trained on large-scale data with various scene layouts and are less robust in novel scenes.

To avoid this issue, our method follows the *detect-first-then-grasp* route. Unlike previous works in this route, our method detects symmetric regions which include both symmetric objects (for objects with global symmetry) and symmetric object regions (for objects with local symmetry). Noteworthily, detecting symmetric regions is more general for grasping, compared to detecting symmetric objects. This is because the former does not require the target object to be perfectly symmetric, and it includes simple primitives that repeat across multiple object types, greatly increasing the cross-instance generalities.

We implement the symmetric region detection by a Mask R-CNN [12] with an FPN backbone. The network is pre-trained on the COCO dataset [19] and fine-tuned on the GraspNet-1-Billion dataset [9]. To collect training data of symmetric regions, we manually annotate the symmetric regions on 12 objects in the GraspNet-1-Billion dataset. The annotations are then propagated to the objects in the RGB-D frames with using the 6D object poses. Our MaskRCNN model achieves an mAP at 0.72 and 0.55 with 0.5 IoU threshold on the seen and unseen subsets, respectively.

E. Symmetry-aware Grasp Detection Network

Next, we describe the architecture and several key considerations of the symmetry-aware grasp detection network.

Principal-directional scale-invariant feature learning. Real-world symmetry annotation is prohibitive to be acquired due to the high labor cost [23]. Existing symmetry detection datasets are annotated by dedicated algorithms and manual adjustment procedures [32]. We propose a learning-based solution to alleviate the problem of insufficient symmetry annotation and increase the cross-instance generality.

The key insight of the solution is: object scaling along some specific directions will not influence the object symmetry. Those include all the directions that are perpendicular or parallel to the normal direction of the reflectional plane and the direction of the rotational axis. Those directions are referred to as *principal directions*. We determine the principal directions of each object based on its symmetry annotation in the canonical coordinate and then transfer them into the camera coordinate using the 6D object pose.

During training, for each detected object, we augment the input points by performing shape scaling on the principal directions and then propose *Principal-directional scale-Invariant Feature Transformer* (PIFT) to learn the feature invariance to scaling along the principal directions. We see this as analogous to the strategies of first adding noise and then denoising in conventional network training.

As shown in Figure 3, taking the cropped point cloud (converted from the RGB-D image) of the detected regions as input, we first augment the initial points by randomly scaling along one of its principal directions. The initial and the augmented point clouds are then fed into a Siamese network respectively for feature extraction. The extracted features are expected to be consistent. To achieve this, we use $\mathcal{L}_2$ loss function to minimize the distance of the two features: $\mathcal{L}^f = \sum_{j=1}^{N} \mathcal{L}_2(f_j^I, f_j^A)$, where f_j^I and f_j^A are the extracted features of the j-th point in the initial and augmented point cloud, respectively, N is the number of points.

The backbone of the Siamese network is a PointNet++ [27] with PIFT modules. The PointNet++ takes N points as input and outputs the point-wise features with the feature-length being 128. The PIFT modules are inserted in front of each set abstraction layer [27]. Formally, given the point cloud P_i

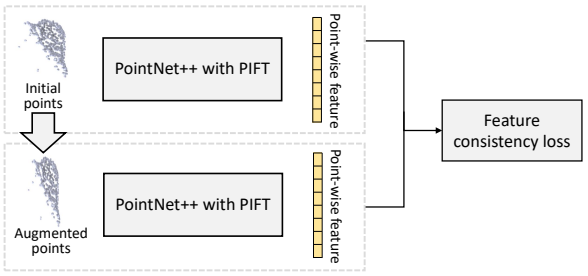

Fig. 3: The feature extraction network is trained in a Siamese fashion. The initial points and the augmented points are passed through a PointNet++ with PIFT modules. A feature consistency loss is adopted to facilitate the learning of principal-directional scale-invariant features.

at the i-th layer and the corresponding point-wise feature E_i, PIFT learns a transformation $T_i = S_i R_i$ that deforms P_i in the camera coordinate to $\widetilde{P}_i$ in a transformed coordinate:

$$\begin{aligned} T_i &= F^{\text{PIFT}}(P_i, E_i), \\ \widetilde{P}_i &= T_i P_i, \end{aligned} \tag{3}$$

where F^{PIFT} is a point convolutional network with a regression layer, $R_i \in \mathbf{R}^{3\times3}$ is the rotation matrix, $S_i \in \mathbf{R}^{3\times3}$ is the scaling matrix. The transformed point cloud $\widetilde{P}_i$ is then fed into a set abstraction layer for feature aggregation:

$$\widetilde{P}_{i+1}, E_{i+1} = F^{\text{SetAbstraction}}(\widetilde{P}_i, E_i), \tag{4}$$

where $F^{\text{SetAbstraction}}$ is the set abstraction layer, E_{i+1} is the extracted feature. The sampled points of the set abstraction layer $\widetilde{P}_{i+1}$ (represented in the transformed coordinate) are then transformed back to the camera coordinate:

$$P_{i+1} = T_i^{-1} \widetilde{P}_{i+1}, \tag{5}$$

where P_{i+1} is the sampled points of the $i+1$-th layer in the camera coordinate. An illustration of the proposed PIFT module is shown in Figure 4.

With such a transformation module, the network is potentially able to predict the canonical coordinate of the point set as well as find optimal local neighborhoods for better feature aggregation. Unlike the typical spatial transformer networks [13] which allow arbitrary affine transformation, our affine transformation T_i is a combination of rotation and scaling, i.e. first rotate the shape into a canonical coordinate, then scale along the axis directions. The PIFT module is a reverse process of the points augmentation mentioned above. That is essentially achieved by learning the principle directions of the input points. Note that the Siamese network is only turned on in the training stage. During testing, we use the trained network to extract features from the initial non-augmented point cloud.

Symmetry and grasp predictions. To estimate the symmetries and detect the grasps, the point-wise features of the initial non-augmented point cloud are fed into several MLPs to make per-point predictions. Each point makes two predictions. First, it predicts the grasp affordance, indicating the confidence the object could be grasped when one finger

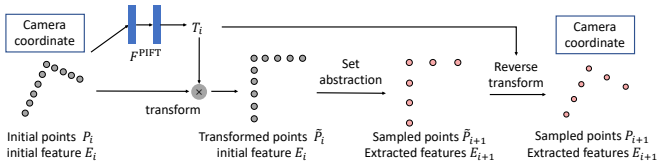

Fig. 4: Principal-directional scale-Invariant Feature Transformer (PIFT) estimates a transformation that allows spatial deformation of points in the point convolution processing, facilitating feature invariance to scaling along principal directions of the symmetry plane/axis.

of the gripper lies on the point. This is achieved by using a regression module. Second, it detects the potential reflectional, rotational, and spherical symmetries. Since the regions in our experiments could have at most 3 reflectional symmetries, 1 rotational symmetry, and 1 spherical symmetry, the network first predicts whether these potential symmetries exist by a classification module and then estimates the parameters of the valid symmetries by a regression module. The network architecture is shown in Figure 2 and Figure 5.

As such, the per-point loss consists of the grasp affordance loss $\mathcal{L}_j^a$ and the symmetry loss $\mathcal{L}_j^s$. To be specific, the affordance loss is:

$$\mathcal{L}_j^a = \mathcal{L}_1(a_j, \hat{a}_j), \tag{6}$$

where a_j and $\hat{a}_j$ are the predicted and ground-truth affordance, respectively. The symmetry loss at the k-th point is:

$$\mathcal{L}_j^s = \sum_{k=1}^{M} [w^c \mathcal{L}_c(c_{jk}, \hat{c}_{jk}) + w^r \mathcal{L}_r(s_{jk}, \hat{s}_{jk})] a_j, \tag{7}$$

where c_{jk} is the estimated probability of the k-th symmetry prediction to be valid. $\hat{c}_{jk}$ is the ground-truth. $\mathcal{L}_c$ is the cross-entropy loss. s_{ij} is the estimated symmetry parameters. $\hat{s}_{ij}$ is the ground-truth. For reflectional symmetry, the estimated parameters include the center of the object and the normal direction of the symmetry plane. For rotational symmetry, the estimated parameters include the center of the object and the axis direction. For spherical symmetry, the estimated parameters only include the center of the object. w^c and w^r are the weights. We set $M = 5$ by assigning 3 output ends for reflectional symmetry estimation, 1 output end for rotational symmetry estimation, and 1 output end for spherical symmetry estimation, as shown in Figure 5.

The parameterizations of s_{jk} are set according to the symmetry type. For reflectional symmetry, s_{jk} contains the center location of the object/region and the normal direction of the mirror plane. For rotational symmetry, it includes the center location and a direction along the rotational axis. $\mathcal{L}_r$ is the dense symmetry loss that penalizes the difference between two symmetries [32], if the symmetry type is rotation or reflection. For spherical symmetry, our network only predicts the center location, so $\mathcal{L}_r$ is the MSE loss.

The symmetry loss is weighed by a regularization term a_j. Intuitively, this means that symmetry estimation will mostly rely on the points with high grasp affordance prediction. This is due to the fact that those points usually lie on the planar

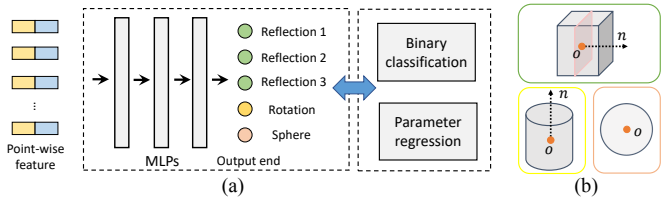

Fig. 5: Representation of symmetry estimation output. (a) The network uses multiple outputs ends to estimate the existence probability and symmetry parameters simultaneously. (b) The parameters of the estimated symmetry are various on different symmetry types.

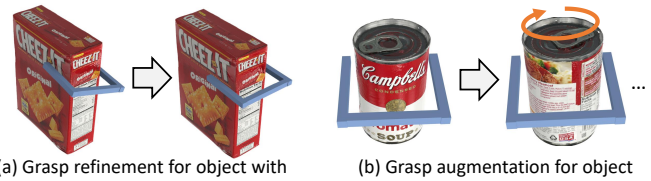

Fig. 6: We refine the annotation of grasp detection by: (a) symmetrizing the grasp poses for objects with reflectional symmetry; (b) duplicating the grasp poses for objects with rotational symmetry.

regions and will be less influenced by the scanning noise, providing more stable features. To sum up, the overall loss is the sum of feature consistency loss, grasp affordance loss and symmetry loss: $\mathcal{L} = w^f \mathcal{L}^f + w^a \sum_{j=1}^{N} \mathcal{L}_j^a + w^s \sum_{j=1}^{N} \mathcal{L}_j^s$, where w^f, w^a and w^s are the weights.

Detecting multiple reflectional symmetries. We use Hungarian algorithm [15] to solve the problem of optimal assignment that matches the symmetry predictions and ground-truth of multiple reflectional symmetries during training. Our method might output multiple reflectional symmetries with high affordance, which needs proper protocol to select the most plausible one to conduct grasp. To achieve this, we compute the normal direction of the target point and that of their predicted counterparts. The symmetry whose corresponding counterpart has the minimum angle difference in the normal direction to the target point is selected as the most plausible one.

Determining grasp points from predicted symmetry. The object symmetry is generated by the per-point symmetry predictions by an averaging operation. For points that have a high estimated grasp affordance (>0.5), we compute their symmetric counterparts. Each point together with its symmetric counterpart represent the contacting points between the object and the gripper. For rotational and spherical symmetries, we only consider the symmetric counterpart with the farthest Euclidean distance as the contacting point.

F. Physics-feasible Grasp Selection

The above network generates graspable points simply based on the local geometry of the symmetric region, which might be physically infeasible or unstable. Therefore, we adopt a grasp selection algorithm to select the plausible ones.

First, we filter the incorrect grasp by checking if the estimated symmetry counterpart (i.e. the grasp point of the gripper) is located in the observed area of the RGB-D frustum. This could happen when the symmetry estimation is incorrect.

Second, we check if the gripper collides with the point cloud of the scene. For each grasp point pair, we sample M grasp candidates via rotating the gripper around the line that connects the two points. The pre-defined gripper model is then transformed into the scene coordinate and conducts a collision detection with the input points. For objects that contain no valid grasp after the above selection protocols, we adopt a lightweight grasp detection approach based on object boundary to generate planar grasps.

Last, we present a heuristic method to rank the grasp candidates. The score of each grasp is computed as:

$$s = \tanh\left(w^b \cdot b\right) \cdot \tanh\left(w^h \cdot h\right), \tag{8}$$

where $b = 5cm$ is the shortest distance between the gripper to the scene points apart from the target region, $h = 3cm$ is the maximum height of scene points inside the gripper in the gripper coordinate, w^b and w^h are hyperparameters. For each object, the detected grasp with the highest score is selected as the optimal grasp.

G. Implementations

Network architecture, parameters. The number of input points is $1,500$. The PointNet++ backbone contains 4 set abstraction layers, each with a PIFT module inserted in front of it, and 4 feature propagation layers. The weights of network training are $w^f = 1$, $w^a = 1$, $w^s = 5$, $w^c = 1$ and $w^r = 5$. The MLPs for symmetry and grasp predictions contain four fully-connected layers. We use Adam optimizer with an initial learning rate of 0.001 which is decreased by 0.9 for every 2 epoch. The batch size is set to be 36 on an NVIDIA Tesla V100 GPU. The training takes about 36 hours to be converged. The inference time is about 0.3 seconds. The hyperparameters in grasp selection are $w^b = 1$ and $w^h = 5$.

Symmetry annotation generation. We first detect the ground-truth symmetry in the canonical coordinate system by the method in [32] and then transfer it into the camera coordinate by using the 6D object pose.

Grasp affordance annotation refinement. Existing high-DoF grasp detection annotations are mostly generated by the sample-and-validate methods [22]. It requires expensive computational costs and is prohibitive to be scaled up. To improve the quality and increase the quantity of the training data, we propose a symmetry-guided grasp affordance annotation enhancement. As shown in Figure 6, we augment the existing grasp detection dataset [9] by two strategies. First, for objects with reflectional symmetries, we refine the grasps by symmetrizing them. Second, for objects with rotational symmetries, we rotate the object and thus duplicate the grasps. These procedures are simple and effective, facilitating more stable network training via resolving the symmetry ambiguity caused by insufficient grasp sampling during the sample-and-validate process.

Fig. 7: Visualization of the detected grasps in cluttered scenes. Our method is able to produce high-quality grasps on objects with complex geometry and heavy occlusion, thanks to the guidance of symmetry estimation.

IV. Results and Evaluation

In this section, we provide details about our experiment setup, results, and evaluations.

A. Datasets & Evaluation Metrics

To evaluate the performance of the proposed method, we conduct experiments on both the existing dataset and a real robot grasping platform. *First*, we evaluate our method on the large-scale GraspNet-1-Billion dataset [9]. The dataset contains $97,280$ RGB-D images from 190 scenes with 88 objects including symmetric objects, such as plier and bottle, as well as non-symmetric objects, such as the printed 3D model of animals. *Second*, we conduct experiments on a real robot grasping platform (Figure 8). The platform contains a UR-5 robot arm with an RG-2 gripper. A realsense camera is on the table to capture the RGB-D images.

We adopt the following evaluation metrics: 1) *Grasp Quality*: An analytic computation method [18] is adopted to compute whether a grasp is plausible, given a friction coefficient. By altering the friction coefficient from $(0, 1]$, we obtain the smallest friction coefficient μ on which the object could be grasped. The grasp quality is then computed as $1 - \mu$. 2) *Antipodal Score*: A simple force closure property of grasps that measures the angle between the normal of the contact points and the line connecting two contact points [28]. The metric is computationally lightweight, which could be applied to real-time grasp evaluation. 3) *Success Rate*: The percentage of objects being successfully grasped by the robot arm. The metric is straightforward and intuitional. It is also sensitive to the gripper type in the robot arm. Since the output of our method is *the most confident grasp on each object* which is different to the problem of detecting all the possible grasps, we did not use the AP metrics [9].

B. Performance on GraspNet-1-Billion

We first compare our method against several baselines on GraspNet-1-Billion. The baselines are selected: 1) Multi-Grasp [4]: A baseline that generates multiple planar grasps in a single shot with a deep neural network. 2) GG-CNN [24]: A baseline that first estimates pixel-level dense grasp affordances and then generates planar grasps with real-time performance. 3) PointNetGPD [18]: The state-of-the-art two-stages 6D grasp detection approach. It first generates grasp candidates with heuristic rules and then adopts a PointNet to evaluate the quality of the candidates. 4) GraspNet [9]: The state-of-the-art 6D grasp detection approach. It generates multiple grasps by a one-stage neural network. All the baselines were trained on GraspNet-1-Billion. While the first two were trained on the planar grasp annotations, the rest were trained on the 6D grasp annotations.

TABLE I: Quantitative comparisons (*Grasp Quality*) to baselines on GraspNet-1-Billion.

Method	**Seen**	**Unseen**
Multi-Grasp [4]	0.061	0.043
GG-CNN [24]	0.078	0.036
PointNetGPD [18]	0.155	0.095
GraspNet [9]	0.291	0.276
Ours	**0.382**	**0.297**

TABLE II: Quantitative comparisons (*Antipodal Score*) to baselines on GraspNet-1-Billion.

Method	**Seen**	**Unseen**
Multi-Grasp [4]	0.182	0.094
GG-CNN [24]	0.254	0.167
PointNetGPD [18]	0.454	0.402
GraspNet [9]	0.731	0.684
Ours	**0.809**	**0.715**

Quantitative Results. We make quantitative comparisons in two evaluation metrics. The results are shown in Table I and Table II, respectively. There are several interesting phenomena we can observe. First, 6D grasp detection approaches are generally better than the planar ones, demonstrating the problem of grasp detection in this dataset is non-trivial. Second, SymmmetryGrasp produces the state-of-the-art performance in terms of *Grasp Quality* on all test sets. This demonstrates its effectiveness and generality. Third, the high *Antipodal Score* of our method indicates that it is able to produce more stable grasps, implying symmetry is indeed helpful.

Qualitative Results. The qualitative results are visualized in Figure 7. It is shown that SymmetryGrasp successfully detects

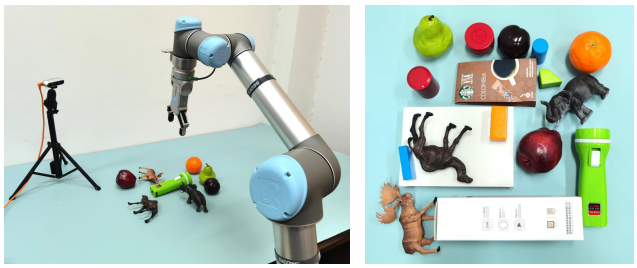

Fig. 8: Left: the experimental robotic grasping platform. Right: the representative objects used in the experiment.

grasps in various scenes. In particular, it produces high-quality grasps on symmetric regions of objects with complex geometry and heavy occlusion.

C. Performance on Real Robot Platform

We also evaluate our method on the real robot platform. For each scene, we use the proposed method to predict the grasps. Then, the robot arm is deployed to perform the grasp with the highest confidence. The task is challenging, as the experimental scenes might contain severe occlusion, various camera viewpoints, and unseen objects. To better reveal the advantages of our method, we divide the objects into several subsets according to their symmetry types: reflection, rotation, sphere, and no-symmetry. Each subset contains 5 objects in various poses with 100 RGB-D images in total being collected. We perform one grasp attempt on each image being collected. In total, there are 400 grasps on all four test subsets. The quantitative comparison is reported in Table III. Note that all methods are trained on GraspNet-1-Billion without any fine-tuning. Our method achieves the best overall performance, providing a more robust solution. The inferior performance of our method on non-symmetric objects reveals the limitation of our method, i.e., it can hardly estimate the contact points on non-symmetric objects with neither global nor local symmetry.

D. Ablation Studies

We conduct several ablation studies to quantify the efficacy of the crucial components. The experimental dataset is GraspNet-1-Billion. Results are reported in Table IV.

No PIFT. We remove the principal-directional scale-invariant feature transformer and retrain the network. This baseline achieves competitive performance compared to the full method on the seen objects, but shows inferiority on the unseen objects. This verifies the need for this component.

No Siamese Training. We turn off the feature consistency loss so there is no Siamese network training component. This baseline learns to estimate symmetry from the initial points and the augmented points separately. Results show the Siamese training scheme is crucial.

No Symmetric Region Segmentation. We conduct an experiment that was only trained to detect and grasp the globally non-symmetric objects. This leads to a slight drop in performance, due to the incapacity of grasping objects, such as the printed 3D models of the animals.

TABLE III: Quantitative comparisons (*Success Rate*) to baselines on the real robot platform.

Method	**Reflection**	**Rotation**	**Sphere**	**Non.**
Multi-Grasp [4]	23%	15%	38%	17%
GG-CNN [24]	27%	28%	35%	33%
PointNetGPD [18]	39%	43%	23%	19%
GraspNet [9]	68%	59%	61%	**52%**
Ours	**74%**	**68%**	**72%**	47%

TABLE IV: Ablation studies. The experiments are conducted on GraspNet-1-Billion. The metric is *Grasp Quality*.

Method	**Seen**	**Unseen**
No PIFT	0.337	0.092
No Siamese Training	0.364	0.148
No symmetric region segmentation	0.350	0.257
No dataset enhancement	0.355	0.240
Full method	**0.382**	**0.297**

No Dataset Refinement. The affordance annotations are optimized by our refinement procedure. We found that without it, the network training tends to be less efficient and the overall performance declines. This is due to the ambiguous affordances caused by insufficient grasp generation sampling.

E. Evaluation on Symmetry Estimation

Symmetry estimation is at the core of our method design. To better illustrate how the estimated symmetry facilities grasp detection, we directly evaluate our symmetry estimation module. We unitize the pre-trained model on GrapsNet-1-Billion to estimate symmetries and evaluate the performance on its test sets. The evaluation method is the average precision [32]. For reflectional symmetry and rotational symmetry, we use the dense symmetry error [32] to determine whether the prediction is true positive. For spherical symmetry, we regard the predictions whose Euclidean distance error is smaller than $5cm$ as true positives. The experiments report the average precision of symmetry estimation at 0.56, 0.61, and 0.73 for the reflection, rotation, and sphere, respectively, on the test set. As comparisons, the state-of-the-art learning-based symmetry detection method [32] achieves 0.39, 0.32, and 0.58, respectively. The recent geometry-based symmetry detection method [7] achieves 0.17, 0.08, and 0.34 respectively. It shows that our method outperform the baselines and is able to accurately detect most of the symmetries despite the shape variations, thanks to the proposed PIFT module.

V. Conclusion

We presented SymmetryGrasp, a learning-based method that detects antipodal grasps from single-view RGB-D images with the guidance of symmetries estimation. We found that grasp detection could greatly benefit from symmetry estimation. In particular, a principal-directional scale-invariant feature transformer is proposed to increase the performance of grasping unseen objects. An interesting future direction is to explore the equivariant networks [37], [6] to enhance the proposed method. We would also like to exploit symmetry in other robotic manipulation tasks, such as object assembling.

REFERENCES

[1] Andrew Blake. A symmetry theory of planar grasp. *The International Journal of Robotics Research*, 14(5):425–444, 1995.

[2] Andrew Blake and Michael Taylor. Planning planar grasps of smooth contours. In *[1993] Proceedings IEEE International Conference on Robotics and Automation*, pages 834–839. IEEE, 1993.

[3] Ch Borst, Max Fischer, and Gerd Hirzinger. A fast and robust grasp planner for arbitrary 3d objects. In *Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No. 99CH36288C)*, volume 3, pages 1890–1896. IEEE, 1999.

[4] Fu-Jen Chu, Ruinian Xu, and Patricio A Vela. Real-world multiobject, multigrasp detection. *IEEE Robotics and Automation Letters*, 3(4):3355–3362, 2018.

[5] Enric Corona, Kaustav Kundu, and Sanja Fidler. Pose estimation for objects with rotational symmetry. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 7215–7222. IEEE, 2018.

[6] Congyue Deng, Or Litany, Yueqi Duan, Adrien Poulenard, Andrea Tagliasacchi, and Leonidas J Guibas. Vector neurons: A general framework for so (3)-equivariant networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12200–12209, 2021.

[7] Aleksandrs Ecins, Cornelia Fermüller, and Yiannis Aloimonos. Seeing behind the scene: using symmetry to reason about objects in cluttered environments. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 7193–7200. IEEE, 2018.

[8] Satoshi Endo, Alan M Wing, and R Martyn Bracewell. Haptic and visual influences on grasp point selection. *Journal of motor behavior*, 43(6):427–431, 2011.

[9] Hao-Shu Fang, Chenxi Wang, Minghao Gou, and Cewu Lu. Graspnet-1billion: A large-scale benchmark for general object grasping. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11444–11453, 2020.

[10] Lin Gao, Ling-Xiao Zhang, Hsien-Yu Meng, Yi-Hui Ren, Yu-Kun Lai, and Leif Kobbelt. Prs-net: Planar reflective symmetry detection net for 3d models. *IEEE Transactions on Visualization and Computer Graphics*, 27(6):3007–3018, 2020.

[11] Minghao Gou, Hao-Shu Fang, Zhanda Zhu, Sheng Xu, Chenxi Wang, and Cewu Lu. Rgb matters: Learning 7-dof grasp poses on monocular rgbd images. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 13459–13466. IEEE, 2021.

[12] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

[13] Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. Spatial transformer networks. *Advances in neural information processing systems*, 28, 2015.

[14] Yun Jiang, Stephen Moseson, and Ashutosh Saxena. Efficient grasping from rgbd images: Learning using a new rectangle representation. In *2011 IEEE International conference on robotics and automation*, pages 3304–3311. IEEE, 2011.

[15] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955.

[16] Susan J Lederman and Alan M Wing. Perceptual judgement, grasp point selection and object symmetry. *Experimental Brain Research*, 152(2):156–165, 2003.

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

[18] Hongzhuo Liang, Xiaojian Ma, Shuang Li, Michael Görner, Song Tang, Bin Fang, Fuchun Sun, and Jianwei Zhang. Pointnetgpd: Detecting grasp configurations from point sets. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 3629–3635. IEEE, 2019.

[19] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[20] Jeffrey Mahler, Jacky Liang, Sherdil Niyaz, Michael Laskey, Richard Doan, Xinyu Liu, Juan Aparicio Ojea, and Ken Goldberg. Dex-net 2.0: Deep learning to plan robust grasps with synthetic point clouds and analytic grasp metrics. *arXiv preprint arXiv:1703.09312*, 2017.

[21] Jeffrey Mahler, Florian T Pokorny, Brian Hou, Melrose Roderick, Michael Laskey, Mathieu Aubry, Kai Kohlhoff, Torsten Kröger, James Kuffner, and Ken 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 *2016 IEEE international conference on robotics and automation (ICRA)*, pages 1957–1964. IEEE, 2016.

[22] Andrew T Miller and Peter K Allen. Graspit! a versatile simulator for robotic grasping. *IEEE Robotics & Automation Magazine*, 11(4):110–122, 2004.

[23] Niloy J Mitra, Mark Pauly, Michael Wand, and Duygu Ceylan. Symmetry in 3d geometry: Extraction and applications. In *Computer Graphics Forum*, volume 32, pages 1–23. Wiley Online Library, 2013.

[24] Douglas Morrison, Peter Corke, and Jürgen Leitner. Closing the loop for robotic grasping: A real-time, generative grasp synthesis approach. *arXiv preprint arXiv:1804.05172*, 2018.

[25] Arsalan Mousavian, Clemens Eppner, and Dieter Fox. 6-dof graspnet: Variational grasp generation for object manipulation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2901–2910, 2019.

[26] Joshua Podolak, Philip Shilane, Aleksey Golovinskiy, Szymon Rusinkiewicz, and Thomas Funkhouser. A planar-reflective symmetry transform for 3d shapes. In *ACM SIGGRAPH 2006 Papers*, pages 549–559. 2006.

[27] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017.

[28] Yuzhe Qin, Rui Chen, Hao Zhu, Meng Song, Jing Xu, and Hao Su. S4g: Amodal single-view single-shot se (3) grasp detection in cluttered scenes. In *Conference on robot learning*, pages 53–65. PMLR, 2020.

[29] Joseph Redmon and Anelia Angelova. Real-time grasp detection using convolutional neural networks. In *2015 IEEE international conference on robotics and automation (ICRA)*, pages 1316–1322. IEEE, 2015.

[30] Joseph Rosen. Symmetry in science. In *Symmetry in Science*, pages 169–183. Springer, 1995.

[31] Yifei Shi, Junwen Huang, Xin Xu, Yifan Zhang, and Kai Xu. Stablepose: Learning 6d object poses from geometrically stable patches. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15222–15231, 2021.

[32] Yifei Shi, Junwen Huang, Hongjia Zhang, Xin Xu, Szymon Rusinkiewicz, and Kai Xu. Symmetrynet: learning to predict reflectional and rotational symmetries of 3d shapes from single-view rgb-d images. *ACM Transactions on Graphics (TOG)*, 39(6):1–14, 2020.

[33] Yifei Shi, Xin Xu, Junhua Xi, Xiaochang Hu, Dewen Hu, and Kai Xu. Learning to detect 3d symmetry from single-view rgb-d images with weak supervision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

[34] Minhyuk Sung, Vladimir G Kim, Roland Angst, and Leonidas Guibas. Data-driven structural priors for shape completion. *ACM Transactions on Graphics (TOG)*, 34(6):1–11, 2015.

[35] Andreas ten Pas, Marcus Gualtieri, Kate Saenko, and Robert Platt. Grasp pose detection in point clouds. *The International Journal of Robotics Research*, 36(13-14):1455–1473, 2017.

[36] Chen Wang, Danfei Xu, Yuke Zhu, Roberto Martín-Martín, Cewu Lu, Li Fei-Fei, and Silvio Savarese. Densefusion: 6d object pose estimation by iterative dense fusion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3343–3352, 2019.

[37] Maurice Weiler and Gabriele Cesa. General e (2)-equivariant steerable cnns. *Advances in Neural Information Processing Systems*, 32, 2019.

[38] Xinchen Yan, Jasmined Hsu, Mohammad Khansari, Yunfei Bai, Arkanath Pathak, Abhinav Gupta, James Davidson, and Honglak Lee. Learning 6-dof grasping interaction via deep geometry-aware 3d representations. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3766–3773. IEEE, 2018.

[39] Andy Zeng, Shuran Song, Kuan-Ting Yu, Elliott Donlon, Francois R Hogan, Maria Bauza, Daolin Ma, Orion Taylor, Melody Liu, Eudald Romo, et al. Robotic pick-and-place of novel objects in clutter with multi-affordance grasping and cross-domain image matching. In *2018 IEEE international conference on robotics and automation (ICRA)*, pages 3750–3757. IEEE, 2018.

[40] Hanbo Zhang, Xuguang Lan, Site Bai, Xinwen Zhou, Zhiqiang Tian, and Nanning Zheng. Roi-based robotic grasp detection for object overlapping scenes. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4768–4775, 2019.

[41] Yu Zheng. An efficient algorithm for a grasp quality measure. *IEEE Transactions on Robotics*, 29(2):579–585, 2012.

[42] Yichao Zhou, Shichen Liu, and Yi Ma. Nerd: Neural 3d reflection symmetry detector. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15940–15949, 2021.

[43] Xinghao Zhu, Lingfeng Sun, Yongxiang Fan, and Masayoshi Tomizuka. 6-dof contrastive grasp proposal network. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6371–6377. IEEE, 2021.