Canonical Shape Reconstruction with SE(3) Equivariance Learning for Weakly-Supervised Object Pose Estimation

Jun Zhou, Kai Chen, Mingqiang Wei, *Senior Member, IEEE*,
Xiao-Ping Zhang, *Fellow, IEEE*, Qi Dou, *Member, IEEE*, and Jing Qin, *Senior Member, IEEE*

***Abstract*—6D object pose estimation from a single RGB-D image is a fundamental problem in computer vision and robot manipulation. Despite recent advancements, existing methods still suffer several limitations. First of all, the object shape representation extracted from the depth map is often less expressive because the object point cloud parsed from the depth map is highly incomplete due to the object self-occlusion and noisy due to the sensor artifacts. This shape representation issue further intensifies when lacking sufficient labeled data for model training, which unfortunately is another typical problem for object pose estimation considering the heavy annotation cost for real-world pose labeling. In this study, we propose to tackle the above issues in a unified way. First, we enhance the object shape representation from the partial point cloud with a novel canonical shape reconstruction module, in which an implicit canonical frame is established by incorporating the SE(3) equivariance, achieving implicit feature alignment of the partial point cloud inputs, leading to robust shape recovery. Second, based on the enhanced object representation, we further utilize the de-canonicalized and pose-dependent completed object shape as the training signal, and develop a novel weakly-supervised learning framework to leverage both labeled synthetic data and unlabeled real data to train the pose estimation model in a label-efficient way. Extensive experiments on three widely used benchmarks demonstrate the effectiveness, and superiority of our framework over state-of-the-art methods.**

***Index Terms*—6D object pose estimation, weakly-supervised training, shape completion in arbitrary poses, SE(3) equivariance.**

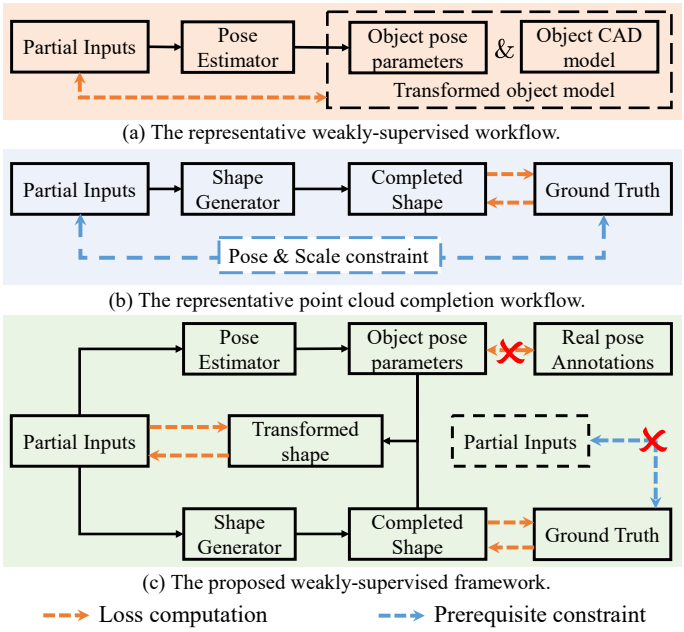


Fig. 1: **Pipeline comparison.** (a) Typical workflow employed by self6d++ [1] and [2]. The transformed object CAD model is used to be a supervised signal for network training. (b) Typical shape completion pipeline. The partial inputs are expected to be aligned with the ground truth in a fixed canonical frame. However, it is an ill-posed problem in the pose estimation task. (c) The proposed weakly-supervised framework. We simultaneously consider pose estimation and shape completion in arbitrary poses. We use the pose-independent completed shape as the supervised signal for network training, showing promising performance.

This work was supported in part by a General Research Fund of Hong Kong Research Grants Council (No. 15218521) and a grant under Theme-based Research Scheme of Hong Kong Research Grants Council (No. T45-401/22-N) and a grant from the Research Grants Council of the Hong Kong (No. 24209223), and the Hong Kong Innovation and Technology Fund (No. ITS/223/22), and the National Natural Science Foundation of China (No. T2322012, No. 62172218) and Shenzhen Ubiquitous Data Enabling Key Lab under grant ZDSYS20220527171406015, and Tsinghua Shenzhen International Graduate School-Shenzhen Pengrui Endowed Professorship Scheme of Shenzhen Pengrui Foundation. (*Corresponding authors: Jing Qin; Kai Chen*)

Jun Zhou and Jing Qin are with the Center of Smart Health, School of Nursing, The Hong Kong Polytechnic University, Hong Kong, China (e-mail: zachary-jun.zhou@connect.polyu.hk; harry.qin@polyu.edu.hk).

Kai Chen and Qi Dou are with the Department of Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong, China (e-mail: kaichen@link.cuhk.edu.hk; qdou@cse.cuhk.edu.hk).

Mingqiang Wei is with the School of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing, China (e-mail: mingqiang.wei@gmail.com).

Xiao-Ping Zhang is with Shenzhen Ubiquitous Data Enabling Key Lab, Shenzhen International Graduate School, Tsinghua University, Shenzhen, China (e-mail: xpzhang@ieee.org).

I. INTRODUCTION

ESTIMATING the 6DoF pose of a known object has been increasingly studied and has been of great importance in many real-world applications both in computer vision and robotics, such as robotic manipulation [3], [4], autonomous driving [5], and 3D scene understanding [6]. Conventional approaches attempt to design handcraft feature descriptors extracted from textures' appearance or geometry structure and recover pose parameters from feature correspondences [7]–

[9]. Although promising progress has been achieved, these methods struggle to work well in challenging scenarios, e.g., varying illumination, severe occlusions, and cluttered backgrounds. Recently, with the prevailing development of RGB-D cameras and neural networks, learning-based approaches with RGB-D data inputs have attracted more attention for their significant performance breakthrough.

However, current methods still suffer from two major limitations. On the one hand, considering the RGB-D data itself, there are often severe artifacts on the depth maps, e.g., outliers and incomplete data, due to the inter-and self-occlusions, as well as reflective surface [10], [11]. Deducing pose parameters from such data is inadvisable and not robust, leading to significant accuracy degradation. One feasible solution is to design advanced feature aggregation methods to get more robust fused RGB-D features for subsequent pose estimation [12]–[14]. Yet, this remedial strategy is also problematic in handling large area data missing. Another promising solution is to adopt shape completion approaches first, which is a more fundamental and untrivial task, and also brings interesting research questions: How to complete the shape and how to use the completed shape? Previous works have proven that completed shapes can facilitate pose estimation and robot grasping process [15]–[17]. Existing point cloud completion methods, however, cannot be employed directly due to limitations imposed by prerequisite conditions: the input partial point cloud is required to be aligned with the ground truth shape within the same canonical reference frame, which means the partial point cloud should have the same pose and scale as the completed shape [18]–[20]. This presents an ill-posed problem in the pose estimation task, as the objective is to obtain the transformation between the coordinate systems of the partial and complete point clouds. Current methods, like [21] and [22], simply normalize the partial point cloud without considering the aligned canonical frame or roughly aligned based on the partial input, resulting in inaccurate shape completion and pose estimation, leaving much room for promoting. Others, like [23] and [24], consider 3D shape reconstruction in the wild relying on the consistency of the multi-view input, while our method performs single-view-based shape reconstruction, showing more efficient and promising performance.

Moreover, these data-driven methods are typically fed with a large amount of annotated data during the training phase, which imposes extreme workloads in pose label annotations. Especially for 6D pose estimation, as it requires each RGB-D frame to be precisely aligned with the CAD models across a cumbersome process [1], [25]. To address the lack of real pose labels, a common line of work simply utilizes rendering tools to generate a huge number of synthetic training images via the projection of known CAD models in random poses [12]–[14], [26]. They also leverage large-scale natural image datasets (e.g., COCO [27]) as rendering background to impose domain randomization [28]. Nonetheless, although more realistic rendering strategies have been made, the performance is still limited compared to methods training with the real pose labels due to the domain gap. Other works [1], [29] initially train the network on large-scale simulated data and then bridge the domain gap by refining the model on unlabeled real data in a self-supervised manner. Yet, the customized simulation procedure and domain gap between synthetic and real data remain bottlenecks for further performance improvement. Recently, [2] introduced a semi-supervised technique for training the network through shape point cloud alignment, which shares some similarities with our approach. However, this technique does not take into account the artifacts present in the depth data and is limited by the absence of a color modality, resulting in performance constraints.

In this paper, we endeavor to tackle these two significant obstacles in a unified way. As illustrated in Fig. 1, we propose a novel weakly-supervised framework for 6D object pose estimation, in which we simultaneously consider shape reconstruction in arbitrary pose and performance improvement without any real pose labels during the training phase. To reconstruct the target shape in different RGB-D frames, we propose canonicalizing the input partial point cloud to a fixed implicit reference frame. Specifically, inspired by [30], taking the mean-centered partial point cloud as input, our reconstruction module first encodes SE(3) equivalent features to disentangle the correlations between the shape geometry representations and the SE(3) pose transformations (e.g., rotation and translation). We then aggregate the instance semantic features with the extracted global invariant shape features, which are fed into a GAN-based generator to reconstruct the completed shape in canonical space. We argue that the implicit reference frame based on the global invariant shape representation can effectively alleviate the feature misalignment problem under arbitrary poses. Furthermore, to train our model without using real pose labels, we develop the independent pose regression branch to compute the pose parameters of the current frame. It is worth noting that the pose and the shape information are disentangled from the input partial point cloud, which means our network can be trained from a natural gradient backpropagation path based on the pose-dependent geometry distance. Concretely, we directly train our model by minimizing the geometry distance between the input partial point cloud and the reconstructed shape transformed by regressed pose parameters. In addition, to bridge the domain gap, we utilize a sim-real joint learning strategy as in [31]. In summary, the main contributions of this work are:

- We propose a unified and effective framework for weakly-supervised 6D object pose estimation, simultaneously tackling shape completion for objects in arbitrary poses and boosting training performance without the need for real pose labels.
- We present an effective shape completion module, in which we leverage the SE(3) equivariance to establish the aligned implicit reference frame. This module is well-suited for pose-variant shape completion and has shown promising results.
- We conduct extensive experiments on three well-acknowledged benchmarks, YCB-Video [32], LineMOD [33], and Occlusion LineMOD [34]. Our method achieves dramatic performance improvements over other state-of-the-art methods without any post-refinement procedures.

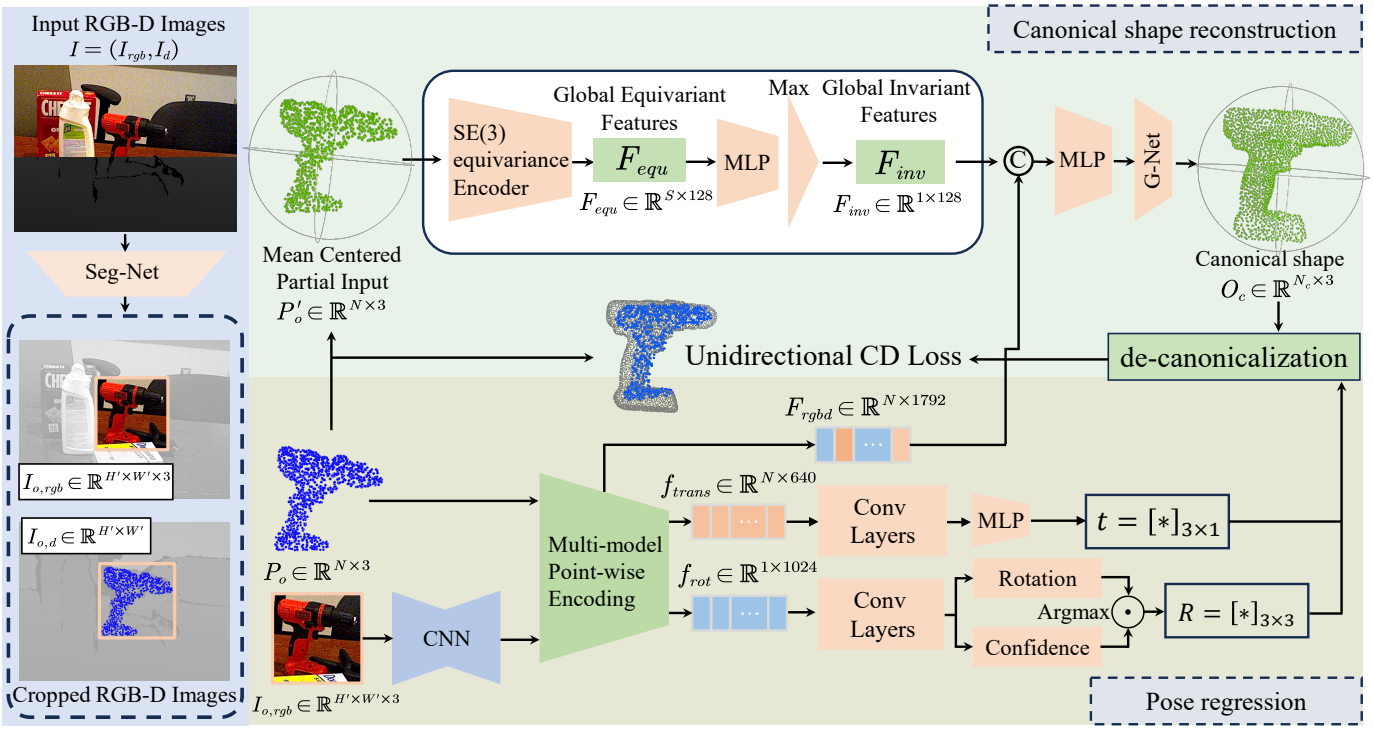


Fig. 2: Overview of our method. The foreground object mask is first obtained by the off-the-shelf instance segmentation network. With the detected mask, our framework takes the cropped image patches as input. For pose estimation, a multi-modality point-wise encoder is employed to obtain the fused RGB-D features and then fed into two separate prediction heads to regress pose parameters. The shape reconstruction module takes the mean-centered partial point cloud as input and generates the global invariant features. By incorporating the fused features, the canonical shape can be recovered from a GAN-based generator. The network is trained by minimizing the geometry distance between the canonical shape and the partial input.

II. RELATED WORK

A. Pose Estimation with Real Pose Annotations.

In this line of work, methods [35]–[40] mainly focus on learning robust color and geometry fused embeddings and then feeding it into prediction heads to regress the final pose parameters directly. Densefusion [12] and its extensions [35], [41] predict pose parameters with a novel pixel-wise dense fusion module, achieving promising results. In these methods, the real pose labels are directly regarded as the supervision signal to update network parameters. Tian et al. [36] propose an improved anchor-based pose regression approach based on dense RGB-D features, boosting the robustness of the model. Another group of methods [13], [14], [42], [43] employs the correspondence-based strategy, in which keypoints on the target object are first detected, and then establish 2D-3D or 3D-3D correspondences by the keypoints feature matching. The final pose results are recovered by using the PnP or Least-Squares Fitting algorithm. In the training phase, the predefined keypoints calculated by the real pose labels are utilized as training signals for the network. Although achieving significant performance improvement, these methods heavily rely on large amounts of real annotated pose labels to train the networks. In contrast, our method can also achieve comparable performance without using real pose labels for training.

B. Pose Estimation without Real Pose Annotations.

In this setting, classical methods [1], [29] mainly employ the coarse-to-fine strategy. Self6D [29] proposes to train the network merely by using the synthetic RGB dataset first in a fully-supervised manner and refining it on unlabeled real data via self-supervision. [2] also follows this scheme but further introduces a new shape-alignment self-learning method when refining the network on the unlabeled real data. DSC-Posenet [44] proposes a two-stage method, which first generates the pseudo masks for the real data and uses the dual-scale consistency loss to predict the keypoints by self-supervised training. Other approaches attempt to annotate the real data automatically. [45] develops an effective self-labeled framework, in which a robot manipulator is utilized to interact with objects in scenarios and annotate the pose labels iteratively. [46] introduces an iterative self-training method, which employs a teacher-student network architecture to collect labels on real data, achieving promising performance. Recently, [31] presents a new sim-real joint-learning method, in which an effective hybrid training loss is designed, showing significant performance improvements in the real data. Inspired by this, our unified weakly-supervised framework can not only complete the partial point cloud accurately at arbitrary pose, but also achieve better performance without real pose labels.

C. Point cloud Completion Approaches

Point cloud completion is a fundamental task in 3D vision. As a classical framework, PCN [18] proposes the first learning-based completion framework, which adopts an encoder-decoder architecture and generates the completed shape from the pooled global representation by FoldingNet [47]. Following this line, several methods [48]–[52] mainly focus on developing hierarchical structures and refinement procedures to pursue detailed completion in higher resolution with better robustness. Recently, PoinTr [19] utilizes the transformer architecture to perform the set-to-set mapping for predicting the missing point proxies. SVDFormer [53] utilizes multiple-view depth images to perceive missing regions, achieving promising performance. However, these methods cannot be directly deployed to target object completion in the pose estimation task, due to their fixed explicit reference frame prerequisites for both partial and completed point cloud. Our method achieves accurate completion performance by utilizing the implicit reference frame to ensure the feature consistency of different partial inputs in arbitrary poses.

III. METHODOLOGY

The goal of 6D object pose estimation is to estimate the 6D pose of a set of known objects in a given RGB-D image of a test scenario. Specifically, the 6D pose is represented by a rigid transformation matrix $T \in SE(3)$, which consists of a rotation $R \in SO(3)$ and a translation $t \in \mathbb{R}^3$. It denotes the relative transformation between the object coordinate system and the camera coordinate system. Different from the fully-supervised approaches, we consider the problem of pose estimation of known objects without real pose labels in this work.

A. Overview

Fig. 2 illustrates the pipeline of our method. We propose a weakly-supervised framework for instance-level object pose estimation. Given an RGB-D image $I = (I_{rgb} \in \mathbb{R}^{H\times W\times 3}, I_d \in \mathbb{R}^{H\times W})$ for a scene, where (H,W) denotes the size of images, we first utilize the object detector to segment each object and obtain the cropped RGB-D image patch $I_O = (I_{o,rgb} \in \mathbb{R}^{H'\times W'\times 3}, I_{o,d} \in \mathbb{R}^{H'\times W'})$ of the target object. With the camera intrinsic matrix, we convert the cropped depth image $I_{o,d}$ into a point cloud $P_o \in \mathbb{R}^{N\times 3}$ as inputs to the subsequent pose estimation network, where N denotes the number of points. Taking the cropped RGB image $I_{o,rgb}$ and the object point cloud P_o as inputs, our multi-modal encoding network extracts color and geometric features in two separate network branches, and then aggregates these cross-modality features densely in a pixel-wise manner. The dense fused features are then fed into two prediction heads for rotation and translation parameters regression. To train our networks in a weakly-supervised fashion without any real pose labels, we further introduce the canonical shape reconstruction module which can rebuild the completed shape of the target object in arbitrary pose. Our shape reconstruction module takes the mean-centered observed object point cloud P'_o as input, utilizing a rotation- and translation-aware encoder to learn the $SE(3)$ equivariant features. By combining the dense pixel-wise fused features and the global invariant features, the canonical shape O_c can be reconstructed by a point cloud generation network, *G-Net*. We first perform the de-canonicalization procedure with the reconstructed canonical shape, i.e., shape size rescaling. Then it is transformed to the new position and orientation O_c^T under the camera coordinate system by using the regressed pose parameters. The training loss is computed between O_c^T and P_o by using the unidirectional chamfer distance to update pose parameters. The details of our framework are described below.

B. Canonical Shape Reconstruction via SE(3) Equivariance Learning.

Most existing shape completion methods are devoted to learning a mapping that predicts a full shape or the rest of structures from the observed partial input. However, this partial-to-completed mapping has a strong precondition. That is, it expects the partial input to be in a fixed canonical coordinate system as same as the completed object shape [19], [51], [52]. Such a premise is contradictory to the pose estimation task since the observed partial object point cloud is indeed in the camera frame. In other words, the 'partial' and 'completed' point cloud have different poses and scales, which are struggling to be handled by existing point cloud completion framework. To overcome this obstacle, we introduce a method to address this ill-posed shape completion problem under arbitrary poses by incorporating SE(3) equivariance. Given a point cloud $X = \{x_i\} \in \mathbb{R}^{N\times 3}$, considering 3D translation $t \in \mathbb{R}^3$ that moves X to X^t. The definition of translation equivariance is:

$$\phi_{f^t}(x_i^t) = \phi_{f^t}(x_i + t) = T_I[\phi_f(x_i)] = \phi_f(x_i), \tag{1}$$

where T_I denotes identity mapping and the feature ϕ_f^t for one point x_i is invariant under any translations. For any $v \in SO(3)$ that rotates X to X^r, the rotation equivariance ensures the extracted features will be transformed according to v. In other words, it satisfies:

$$\phi_{f^r}(x_i^r) = \phi_{f^r}(v \cdot x_i) = v \cdot T_I[\phi_f(x_i)] = v \cdot \phi_f(x_i), \tag{2}$$

By integrating these two equivariance, we can get a rotation- and translation-equivariant transformation T_δ in $G \in SE(3)$, where G is a subgroup of SE(3) since the SO(3) rotation space is used to discretized into the icosahedron rotation group for a discrete approximation particularly [54], [55]. Intuitively, given an observed partial point cloud with an arbitrary pose $v_i = [R_i|t_i] \in \mathcal{V}$, where $\mathcal{V}$ is a set of pose matrix in a fixed frame, we can obtain consistent equivariant features by means of SE(3) equivariance encoding without considering the effect of feature space distortion caused by rotation.

Having the SE(3)-equivariant representation is crucial for the subsequent reconstruction of the canonical shape. In order to extract the SE(3)-equivariant features from the point cloud in arbitrary poses, we first normalize the observed partial point cloud P_o in a unit sphere with the mean-centered coordinates $(P_o \rightarrow P'_o)$. Based on the normalized shape coordinates, we further extract the pose-independent implicit geometry embeddings for shape representation. Specifically, without loss of generality, we resort to the classic Tensor Field Networks

(TFN) [56], [57] to implement our backbone network Φ to extract SE(3)-equivariant features $F_{equ} \in \mathbb{R}^{S\times C} = \Phi(P'_o)$, where S denotes number of Fibonacci sphere sampling and C is channel dimension (S, C are set to 64 and 128 in our experiments respectively). The TFN is a general feature learning neural network for 3D point clouds that leverages tensor representations to capture geometric and topological information. It maps the point cloud to feature space under the constraint of SE(3)-equivariance, i.e., outputs the rotationally equivariant and translationally invariant features, which means the TFN is locally equivariant to 3D rotations, translations, and permutations of the 3D points at each layer. In computation, the TFN takes the point cloud $X = \{x_i\} \in \mathbb{R}^{N\times 3}$ as input, the output of each layer at x_i is given by: $F^{\ell}_{out,i} = \sum_{k\geq 0}\sum_{j=1}^{n}(W^{\ell k}(x_j - x_i)F^{k}_{in,j})$, where $\ell, k \in \mathbb{N}$ is the rotation order (aka type), $W^{\ell k} : \mathbb{R}^3 \rightarrow \mathbb{R}^{(2\ell+1)\times(2k+1)}$ is a learnable weight kernel, mapping the k-rotation-order features to ℓ-rotation-order features. The produced global features $F^{\ell}_{out,i}$ satisfy the equivariance property. In this way, we can map the observed object point cloud in arbitrary poses to an equivariant feature space, which serves as the foundation for the subsequent shape reconstruction procedure. For more details, please refer to [56]–[59]. Our goal is to reconstruct the canonical shape of the target object, which means that the reconstructed shape should be invariant to arbitrary SE(3) transformation of the input P'_o. To this end, we propose to learn a function mapping φ that generates a SE(3)-invariant feature F_{inv}, i.e., $\varphi : F_{equ} \mapsto F_{inv}$, and we have:

$$F_{inv} = \varphi(\Phi(P'_o)) = \varphi(\Phi(T_{arb}[P'_o])),\ \forall\ T_{arb} \in SE(3), \tag{3}$$

Specifically, we propose to implement φ with an MLP-based architecture, and then employ a max-pooling layer to obtain a global SE(3)- and permutation-invariant feature F_{inv}:

$$F_{inv} \in \mathbb{R}^{1\times C} = MAX(MLPs(\Phi(P'_o))), \tag{4}$$

where MAX denotes the max-pooling operation. Afterward, we employ the invariant feature F_{inv} as an implicit representation for shape consistency, which facilitates the subsequent canonical shape reconstruction process. Concretely, to enhance shape feature representation, we first concatenate the fused RGBD features F_{rgbd} (see Sec. III-C) with the invariant feature F_{inv}. After passing through two fully connected layers, we employ a point cloud generation network, G-Net, which specifically leverages the generator architecture from tree-GAN [60]. Tree-GAN is a GAN-based network designed for 3D point cloud generation, comprising a graph convolutional generator and a discriminator, akin to the architecture utilized in r-GAN [61]. G-Net generates the canonical shape $O_c \in \mathbb{R}^{N_c\times 3}$ by taking these fused features as input, as formulated below:

$$O_c = G\text{-}Net(MLP(F_{rgbd} \oplus F_{inv})), \tag{5}$$

C. Object Pose Estimation with Dense Aggregated RGBD Features

Taking the preprocessed object image patch $I_{o,rgb} \in \mathbb{R}^{\hat{H}\times\hat{W}\times 3}$ and the observed object point cloud $P_o \in \mathbb{R}^{N\times 3}$ as input, we establish an end-to-end with two separate feature extraction

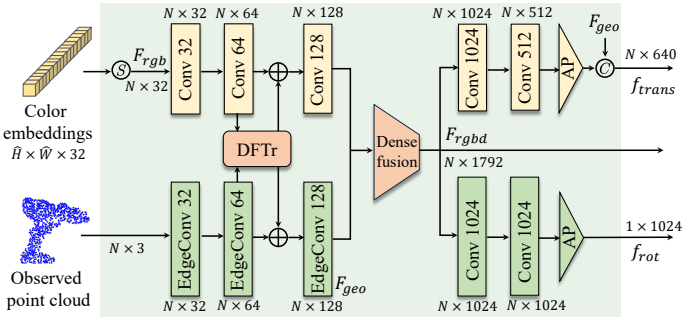


Fig. 3: The architecture of color and geometry feature encoding. The DFTr [14] and Densefusion [12] blocks are employed for RGB-D feature aggregation.

branches (i.e., color and geometric) network to predict rotation R and translation t independently. Concretely, we use a fully convolutional architecture based on PSPNet [62] to extract the appearance features, in which a pre-trained ResNet34 [63] is used as the encoder with the pyramid pooling modules in the last encoding layer. Then three-level upsampling modules with a final convolutional layer are utilized as the decoder. For geometric feature learning, three-level edge convolution operation (EdgeConv) proposed by DGCNN [64] are employed recurrently to map each point of the P_o into a D-dimensional feature space. Between these two branches, we employ the deep fusion transformer (DFTr) block proposed by [14] for better cross-modality features aggregation, and then utilize the densefusion module [12] to get dense fused RGBD features F_{rgbd} for subsequent pose prediction, as shown in Fig. 3.

Given the dense fused RGBD features F_{rgbd}, we utilize two prediction heads to regress the rotation and translation parameters respectively. Considering that the translation is a dense prediction task, we insert additional geometric features F_{geo} into the translation prediction head to enhance the representation of spatial geometry information, and concatenate it with the high-level dense fused RGBD features, as in [36]. Then we propose to regress the transformation matrix between the canonical frame and the camera frame directly by using MLP-liked architectures. Let $\mathcal{T}^{can}_{cam}$ be the transformation from the camera frame to the canonical frame. Thus we have:

$$\mathcal{T}^{can}_{cam} = \mathcal{T}^{obj}_{can} + \mathcal{T}^{obj}_{cam} = PRED_t(f_{trans}), \tag{6}$$

where $PRED_t(\cdot)$ is the prediction head, $\mathcal{T}^{obj}_{can}$ denotes the transformation from canonical frame to object frame obtained in object model canonicalization process, and $\mathcal{T}^{obj}_{cam}$ is obtained from the rendering process. To alleviate the problem of non-linearity of rotation space [26] and the issue of local optimum during the training process caused by symmetric objects [65], we employ an anchor-based approach proposed by [36] for rotation regression. Specifically, we utilize the icosahedral group (N=60) [66] to sample the whole $SO(3)$ space uniformly, where N is the number of anchors. Then the rotation branch predicts a deviation ΔR^{θ}_i for each rotation anchor R^*_i. The completed rotation matrix R_i for i-th anchor is computed by:

$$R_i = \Delta R^{\theta}_i \cdot R^*_i, \tag{7}$$

We leverage the unit quaternion (4D) for rotation representation [67]. For each predicted rotation hypothesis R_i, we use an additional head to output a confidence score $c_i \in [0,1]$. The hypothesis with the highest confidence score will be selected as the final rotation parameter during the inference phase.

D. Weakly-Supervised Training and Framework Inference

Unlabeled real data training. Given the reconstructed canonical shape O_c, we train our network in a weakly-supervised manner without any real pose labels. Note that O_c has a fixed initialized pose information T_p^0 as same as the known object CAD model, which means we can utilize the deviation pose ΔT_i between T_p^0 and the target pose T_p^{tar} to backpropagate gradient. Consequently, we directly calculate geometry distance loss between the transformed canonical shape and the observed partial input point cloud to train our network. Specifically, we first rescale the reconstructed canonical shape for de-canonicalization, and then transform it into the camera frame by the regressed rotation R_{reg} and translation t_{reg}. Due to the incompleteness of the observed input P_o, evaluating the consistency between these two point clouds directly cannot accurately reflect the quality of the predicted pose. To mitigate this issue, we employ the unidirectional chamfer distance (UCD) as in [68] to compute the geometry distance, enforcing each point in P_o finds a corresponding point in the transformed canonical shape O_c^T. Formally, we have:

$$\mathcal{L}_{geo} = \mathcal{d}_{UCD}(P_o, O_c^T) = \frac{1}{|P_o|} \sum_{p_i \in P_o} \min_{p_j \in O_c^T} ||p_i - p_j||^2, \quad (8)$$

where $O_c^T = sR_{reg}O_c + t_{reg}$, s denotes the normalization scale obtained from the object CAD model. In the canonical shape reconstruction branch, we use the normalized object CAD model O_c^* for supervision. The normalized model is centered at the origin and scaled to fit within the unit sphere, as described in [69], [70]. Concretely, to improve the reconstruction quality of the canonical shape, we leverage the density-aware chamfer distance (DCD) [71], which effectively resolves the issue of non-uniform point cloud density inherent in standard CD metric, formulated as follows:

$$\mathcal{L}_{rec} = \mathcal{d}_{DCD}(O_c, O_c^*) = \frac{1}{2}\left(\frac{1}{|O_c|}\sum_{x \in O_c}(1 - \frac{e^{Z_x}}{n_{\hat{y}}}) + \frac{1}{|O_c^*|}\sum_{y \in O_c^*}(1 - \frac{e^{Z_y}}{n_{\hat{x}}})\right) \quad (9)$$

where $Z_x = -\alpha||x - \hat{y}||_2$, $Z_y = -\alpha||y - \hat{x}||_2$. α is a temperature scalar and $\hat{x}(\hat{y}) = min_{x(y) \in O_c(O_c^*)}||x(y) - y(x)||_2$, for more details, please refer [71]. For real data training loss, we have:

$$\mathcal{L}_{real} = \mathcal{L}_{geo} + \mathcal{L}_{rec}, \quad (10)$$

To train our network in a more stable way, in the training phase, we first pretrain our pose estimation branch on the synthetic dataset and then utilize a sim-real joint learning strategy as in [31] to train our entire model.

Synthetic data training. Specifically, we supervise the rotation and translation prediction separately, formulated as follows:

$$\mathcal{L}_{rot} = \sum_{i=1}^{N}\left(\frac{L_i}{d} \cdot c_i - w \cdot log(c_i)\right), \quad \mathcal{L}_{tra} = ||t_{reg} - t_{reg}^*||, \quad (11)$$

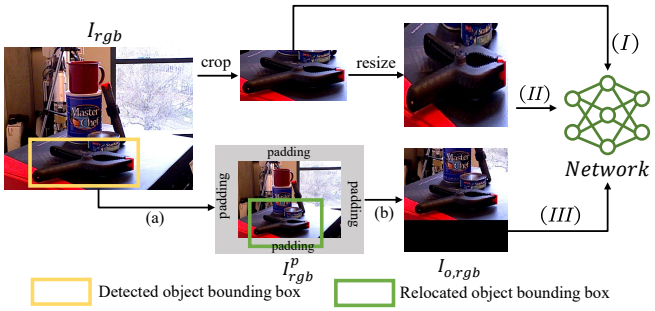


Fig. 4: Illustration of the image patch preprocessing. (a) Image padding. (b) Cropped based on the relocated bounding box. Type of cropping: (I) Without resizing. (II) Resizing directly. (III) Ours.

where L_i denotes the ShapeMatch-Loss [36], [65] of each R_i, d is the object diameter, w is hyperparameter, t_{reg}^* is the ground truth translation. Moreover, a regularization loss [36] is employed to constrain the orthonormality of R_i, i.e., $\mathcal{L}_{ort} = \sum_{i=1}^{N} max(0, \max_{j \neq i}\langle q_i, \hat{q}_j\rangle - \langle q_i, \hat{q}_i\rangle)$. The total synthetic data training loss is formulated as:

$$\mathcal{L}_{syn} = \lambda_1\mathcal{L}_{rot} + \lambda_2\mathcal{L}_{tra} + \lambda_3\mathcal{L}_{ort}, \quad (12)$$

where λ_{1-3} denotes the balance hyperparameters, we set $\lambda_1 = 1.0$, $\lambda_2 = 5.0$, $\lambda_3 = 2.0$ in our experiments.

Overall loss function. The total sim-real joint training loss in each batch is formulated as:

$$\mathcal{L} = \frac{1}{N_{real}}\sum_{i=1}^{N_{real}}\mathcal{L}_{real}^i + \frac{1}{N_{syn}}\sum_{j=1}^{N_{syn}}\mathcal{L}_{syn}^j, \quad (13)$$

where N_{real} and N_{syn} are the number of training samples in each batch. Note that $\mathcal{L}_{real}$ and $\mathcal{L}_{syn}$ only work on corresponding data (real and synthetic) respectively in training stage.

Framework inference. Upon completion of training, given the RGB-D images of the test scene, our model will produce three outputs, i.e., the pose parameters (rotation R_{reg} and translation t_{reg}) of the target object from the pose estimation branch, and the reconstructed complete shape of the target object from the canonical shape reconstruction branch. The acquisition of the pose parameters is our ultimate objective, while the reconstructed shape of the target object can be regarded as an ancillary output of our framework.

E. Implementation and Training Details

Similar to other segmentation-driven methods [72], the instance segmentation and the subsequent pose estimation are disentangled. We first segment the objects of interest in the image by using an off-the-shelf network (e.g., Segment Anything [73]). After that, we crop the target object from the RGB-D image based on the segmentation results. However, the size of the cropped image patches varies due to the shape of each object, which will bring about some complications for the network. The first problem is that the training batch size is limited to 1, which would hinder the performance of the network and bring instability to the training process. Another

one is that the training speed would be restricted greatly due to the batch size.

To address these obstacles, a naive solution is to directly reshape these patches to a fixed size (e.g., 192×192), but such a process would cause the object to undergo dramatic deformation and degrade the performance of the network. To this end, as shown in Fig. 4, we propose to utilize a center-fixed strategy to maintain the contour of objects during the resizing process. Specifically, let $b = (c_{min}, c_{max}, r_{min}, r_{max})$ be the detected bounding box on the raw image. We first pad the raw image based on its longer edge and then compute new coordinates b' of b in the padded image I^p_{rgb}, which are as follows:

$$\hat{c}_{min(max)} = bb_{cx} + \mathbb{I} \cdot \frac{\mathcal{S}\hat{W}}{2 \cdot L_{min}}, \hat{r}_{min(max)} = bb_{cy} + \mathbb{I} \cdot \frac{\mathcal{S}\hat{H}}{2 \cdot L_{min}}, \tag{14}$$

where bb_{cx} and bb_{cy} denote x, y coordinates of the bounding box center in I^p_{rgb}, $\hat{W}$ and $\hat{H}$ denote the target size ($\hat{W} = \hat{H} = 192$ in our experiments). $\mathbb{I}$ = 1 if $\hat{c}_{min}(\hat{r}_{min})$, else -1. $\mathcal{S}$ is a crop parameter that is the minimum value between length of the longest edge of b' and the distance between the object center and the boundary of I^p_{rgb}. L_{min} is the larger value between $\hat{W}$ and $\hat{H}$. With the equation (14), we can ensure that the object center always aligns with the image center, thereby achieving non-deformation of the bounding box during the resize operation. We will show that our straightforward yet effective strategy yields noticeable enhancements in both accuracy and training speed in the experiments.

The size of the input RGB image is $480 \times 640 \times 3$. For each object, we reshape the cropped image patch to 192×192 by our algorithm and randomly sample 1024 points on the depth map as the input of our framework. The number of points N_c for the generated canonical shape O_c is set to 2048. We exclusively employ the pretrained ResNet34 [74] in our CNN encoder during training, without utilizing any other pretrained models. To ensure a stable training process, we first pre-train our pose prediction network on the synthetic dataset for 15 epochs with the supervised loss $\mathcal{L}_{syn}$ and then jointly optimize the entire model with the unlabeled real data for 30 epochs. In Eq.13, For each batch, we sample 16 synthetic samples and 4 real samples. The network is trained by the Adam optimizer with the cyclical learning rates adjustment schedule [75] on two NVIDIA 3090 GPUs, taking approximately 35 epochs (about 5 hours) to reach optimal performance for each category.

IV. EXPERIMENTS AND RESULTS

A. Experiments Settings

Datasets. We evaluate our method on three widely used benchmark datasets, i.e., **YCB-Video** [32], **LineMOD** [33] and **Occlusion LINEMOD** [34]. Specifically, YCB-Video [32] contains 21 objects with various textures and shapes, and the well-annotated RGB-D images are captured from 92 videos in varying scenes. We follow prior works [13], [14], [65] to split the training/testing set. LineMOD [33] includes 13 low-textured objects with diverse shape structures. The texture-less surfaces and varying illumination as well as the cluttered scenario make quite a challenge. The standard training and testing set is split as in [26], [65]. Occlusion LINEMOD [34] contains a subset of LINEMOD datasets collected by additionally annotating. The multi-objects with heavy occlusion in the complex scenes make the pose estimation a great challenge. For a fair comparison, we use the physically-based rendered (PBR) synthetic and unlabeled real data from the BOP benchmark [80] for network training, in alignment with existing methods.

Evaluation Metrics. We use two standard metrics to evaluate our method (i.e., Average Distance of Model Points (ADD) [81] and Average Closest Point Distance (ADD-S) [65], designed for asymmetric and symmetric objects respectively. Concretely, the ADD (ADD-S) metric computes the mean (closest point) distance between the object point sets transformed by the estimated pose and the ground truth pose respectively. For the YCB-Video datasets, we report the area under the ADD-S and ADD(S) curve (ADD-S and ADD(S) AUC) and the ADD-S smaller than 2 centimeters (ADD-S<2cm) following [12], [13], [65]. For the LineMOD and Occlusion LineMOD datasets, we report the ADD distance less than 10% of the object's diameter (ADD-0.1d) as in [26].

B. Comparison With State-of-The-Art Methods.

Performance on the YCB-Video dataset. In Tab. I, we first evaluate our method on YCB-V and compare it with other state-of-the-art methods. Our method achieves better performance compared with self6D++ [1] and makes 1.2% and 3.0% improvements on ADDS and ADD(S) metrics respectively in terms of average accuracy. Since there are few evaluation results on the YCB-V dataset, we also compare our method with other RGB-D based methods trained with real pose labels. In particular, our model surpasses Densefusion [12] on both three metrics (i.e., ADDS, ADD(S), and ADD-S<2cm) and achieves comparable performance compared with other approaches, without any usage of post-refinement procedure (e.g., ICP [82]). These experimental results demonstrate the effectiveness of our proposed network. Moreover, we provide qualitative comparison results on the YCB-V dataset in Fig. 5a, in which our method performs better performances with robustness to occlusion both for symmetric and texture-less objects, demonstrating its effectiveness.

Performance on the LineMOD dataset. We then evaluate our method on the LineMOD dataset in Tab. II. Overall, our proposed method significantly outperforms other existing methods trained on unlabeled real data, achieving the best performance (96.0%) in terms of average accuracy. Specifically, our model achieves 7.5% and 1.8% performance improvement compared with self6D++ [1] and [2] respectively. In addition, compared with the latest RGB-based approaches, our model surpasses SMOC-Net [77], TexPose [78] and [79] by 4.7%, 4.3% and 3.8% respectively. For comparison with the fully-supervised approach, our proposed method also outperforms Densefusion [12] and NVR [40] by 2.7% and 1.7% respectively, showing promising performance.

Performance on the Occlusion LineMOD dataset. To analyze the robustness of our method against severe occlusion,

TABLE I: Quantitative comparison results without post-refinement on the YCB-Video benchmark. We report the ADD-S AUC, ADD(S) AUC, and ADDS <2cm metrics. Symmetric objects are in bold. DF (per-pixel) means DenseFusion (per-pixel) [12].

Supervision	w/ Real Pose Labels									w/o Real Pose Labels						
Methods	DF (per-pixel) [12]			PVN3D [42]			FFB6D [13]			Self6D++ [1]		Lin et al. [76]	Zhou et al. [2]	Ours		
	ADDS	ADD(S)	<2cm	ADDS	ADD(S)	<2cm	ADDS	ADD(S)	<2cm	ADDS	ADD(S)	ADD(S)	ADDS	ADDS	ADD(S)	<2cm
002 master chef can	95.2	70.7	100.0	96.0	80.5	100.0	96.3	80.6	100.0	88.8	8.4	-	95.4	91.0	68.4	100.0
003 cracker box	92.5	86.9	99.3	96.1	94.8	100.0	96.3	94.6	97.0	94.2	84.9	-	88.6	87.3	69.9	89.3
004 sugar box	95.1	90.8	100.0	97.4	96.3	100.0	97.6	96.6	100.0	95.8	88.0	-	95.3	95.6	87.9	100.0
005 tomato soup can	93.7	84.7	96.9	96.2	88.5	98.1	95.6	89.6	97.2	90.8	79.4	-	93.6	94.2	79.2	99.2
006 mustard bottle	95.9	90.9	100.0	97.5	96.2	100.0	97.8	97.0	98.9	98.6	92.7	95.8	96.8	93.2	83.9	100.0
007 tuna fish can	94.9	79.6	100.0	96.0	89.3	100.0	96.8	88.9	100.0	97.5	89.7	90.4	96.2	94.2	78.7	100.0
008 pudding box	94.7	89.3	100.0	97.1	95.7	100.0	97.1	94.6	100.0	98.4	93.9	-	89.7	92.3	81.0	100.0
009 gelatin box	95.8	95.8	100.0	97.7	96.1	100.0	98.1	96.9	100.0	94.0	83.9	-	96.0	97.3	94.1	100.0
010 potted meat can	90.1	79.6	93.1	93.3	88.6	94.6	94.7	88.1	97.9	89.3	75.7	-	90.1	86.9	68.0	96.4
011 banana	91.5	76.7	93.9	96.6	93.7	100.0	97.2	94.9	96.1	98.5	91.8	52.5	93.2	85.6	68.8	76.0
019 pitcher base	94.6	87.1	100.0	97.4	96.5	100.0	97.6	96.9	100.0	98.9	92.1	-	96.6	90.2	74.5	100.0
021 bleach cleanser	94.3	87.5	99.8	96.0	93.2	100.0	96.8	94.8	98.2	93.5	84.5	-	92.0	93.9	81.7	100.0
024 bowl	86.6	86.6	69.5	90.2	90.2	80.5	96.3	96.3	78.8	89.1	89.1	-	87.4	84.8	84.8	72.9
025 mug	95.5	83.8	100.0	97.6	95.4	100.0	97.3	94.2	100.0	94.1	81.4	65.6	96.7	93.8	86.1	100.0
035 power drill	92.4	83.7	97.1	96.7	95.1	100.0	97.2	95.9	98.2	95.2	84.2	81.0	91.6	94.6	88.0	100.0
036 wood block	85.5	85.5	93.4	90.4	90.4	93.8	92.6	92.6	99.4	78.3	78.3	-	89.8	92.1	92.1	99.2
037 scissors	96.4	77.4	100.0	96.7	92.7	100.0	97.7	95.7	98.3	69.2	45.2	-	81.7	95.6	89.7	100.0
040 large marker	94.7	89.1	99.2	96.7	91.8	99.8	96.6	89.1	100.0	87.5	74.6	-	97.3	95.9	85.9	100.0
051 large clamp	71.6	71.6	78.5	93.6	93.6	93.6	96.8	96.8	100.0	79.2	79.2	-	72.0	95.5	95.5	100.0
052 extra large clamp	69.0	69.0	69.5	88.4	88.4	83.6	96.0	96.0	99.2	87.3	87.3	-	65.7	91.0	91.0	99.4
061 foam brick	92.4	92.4	100.0	96.8	96.8	100.0	97.3	97.3	100.0	95.5	95.5	-	93.9	94.1	94.1	100.0
ALL	91.2	82.9	95.3	95.5	91.8	97.6	96.6	92.7	98.1	91.1	80.0	-	90.9	**92.3**	**83.0**	**96.8**

TABLE II: Quantitative evaluation in terms of ADD(S)-0.1d metric on the LineMOD dataset. Symmetric objects are in bold. * denotes RGB only based method. † denotes depth (D) only based method.

Supervision	w/ Real Pose Labels				w/o Real Pose Labels								
Training data	RGB-D				RGB				RGB-D				
Methods	NVR [40]*	DenseFusion [12]	FFB6D [13]	DFTr-Net [14]	DSC [44]	SMOC-Net [77]	TexPose [78]	Hai et al. [79]	Self6D [29]	Lin et al. [76]	Self6D++ [1]	zhou et al. [2]†	Ours
ape	83.3	92.3	98.4	98.6	31.2	85.6	80.9	81.9	38.9	67.5	75.4	91.7	**97.4**
benchvise	98.8	93.2	100.0	100.0	83.0	96.7	99.0	95.0	75.2	**99.9**	94.9	95.8	**99.9**
camera	94.9	94.4	99.9	100.0	49.6	**97.2**	94.8	94.2	36.9	87.4	97.0	96.7	95.3
can	98.2	93.1	99.8	100.0	56.5	**99.9**	99.7	96.8	65.6	99.2	99.5	88.8	94.4
cat	95.4	96.5	99.9	100.0	57.9	95.0	92.6	95.4	57.9	94.3	86.6	99.1	**99.8**
driller	98.3	87.0	100.0	100.0	73.7	**100.0**	97.4	94.8	67.0	97.6	98.9	97.3	99.5
duck	85.2	92.3	98.4	99.1	31.3	76.0	83.4	83.5	19.6	67.2	68.3	80.8	**92.0**
eggbox	99.9	99.8	100.0	100.0	96.0	98.3	94.9	93.9	99.0	98.9	99.0	**100.0**	94.5
glue	99.6	100.0	100.0	100.0	63.4	99.2	93.4	96.5	94.1	96.2	96.1	**100.0**	**100.0**
holepuncher	91.5	92.1	99.8	100.0	38.8	45.6	79.3	84.5	16.2	49.9	41.9	94.4	**97.5**
iron	98.6	97.0	99.9	99.9	61.9	**99.9**	99.8	94.9	77.9	99.5	99.4	88.8	99.8
lamp	99.6	95.3	99.9	100.0	64.7	98.9	98.3	94.8	68.2	**99.8**	98.9	96.7	98.1
phone	94.8	92.8	99.7	99.6	54.4	94.0	78.9	94.1	50.1	91.5	94.3	**96.1**	92.3
MEAN	95.3	94.3	99.7	99.8	58.6	91.3	91.7	92.2	58.9	88.4	88.5	94.2	**97.0**

we further evaluate it on the Occlusion LineMOD dataset, and present the quantitive comparison of the state-of-the-art methods and our network. As shown in Tab. III, our method consistently achieves the best mean performance in terms of ADD(S) AUC (67.4%) compared with other existing methods. Concretely, our model advances self6d++ [29] and [2] by 2.7% and 11.6% respectively, and achieves 0.7%, 2.0% and 4.1% accuracy improvements compared with the RGB-based method TexPose [78], [79] and SMOC-Net [77]. In particular, our approach outperforms the fully-supervised methods NVR [40], DGECN++ [43] and FFB6D [13] by 10.8%, 7.5% and 1.2% respectively, highlighting its superiority and robustness towards occlusion. Qualitative comparison results in Fig. 5b also demonstrate the effectiveness of our approach, and substantiate the superiority of our method in robustly handling varying occlusions compared with self6d++ [29]. We argue that the decoupling of shape and pose information and precise shape reconstruction can effectively alleviate the ambiguity of pose perception in the local-to-global pose prediction process.

C. Ablation Studies.

To justify the design choices of our method, we conduct comprehensive ablation studies to explore the influence of individual components.

Effect of image patch processing algorithm. To investigate the influence of the image patch process method employed by our framework, we conduct comprehensive comparison experiments on the LineMOD dataset. In detail, we compare

TABLE III: Quantitative comparison of ADD(S)-0.1d on the Occlusion-LineMOD benchmark. Symmetric objects are in bold. * denotes RGB only based method. † denotes depth (D) only based method. Our method achieves the SOTA performance in terms of average accuracy.

Supervision	w/ Real Pose Labels					w/o Real Pose Labels								
Training data	RGB-D					RGB				RGB-D				
Methods	NVR [40]*	DGECN++ [43]	PVN3D [42]	FFB6D [13]	DFTr-Net [14]	DSC [44]	SMOC-Net [77]	TexPose [78]	Hai et al. [79]	Self6D [29]	Lin et al. [76]	Self6D++ [1]	zhou et al. [2]†	Ours
ape	43.1	52.1	33.9	47.2	64.1	9.1	60.0	60.5	60.1	13.7	40.3	59.4	50.4	**64.2**
can	82.9	76.3	88.6	85.2	96.1	21.1	94.5	93.4	94.2	43.2	75.2	**96.5**	49.2	84.1
cat	27.2	27.5	39.1	45.7	52.2	26.0	59.1	56.1	56.5	18.7	35.0	**60.8**	30.2	40.2
driller	69.7	78.3	78.4	81.4	95.8	33.5	**93.0**	92.5	89.7	32.5	68.5	92.0	59.5	86.1
duck	44.2	55.2	41.9	53.9	72.3	12.2	37.2	55.5	30.9	14.4	25.7	30.6	40.6	**57.3**
eggbox	49.7	62.3	80.9	70.2	75.3	39.4	48.3	46.0	58.1	57.8	44.7	51.1	**76.3**	61.4
glue	74.3	66.6	68.1	60.1	79.3	37.0	**89.3**	82.8	88.9	54.3	60.7	88.6	75.9	81.3
holepuncher	61.7	60.6	74.7	85.9	86.8	20.4	25.0	46.5	44.2	22.0	28.0	38.5	**68.6**	64.2
MEAN	56.6	59.9	63.2	66.2	77.7	24.8	63.3	66.7	65.4	32.1	47.3	64.7	55.8	**67.4**

TABLE IV: Effect of components of the loss functions in terms of ADD(S)-0.1d metric on LineMOD benchmark.

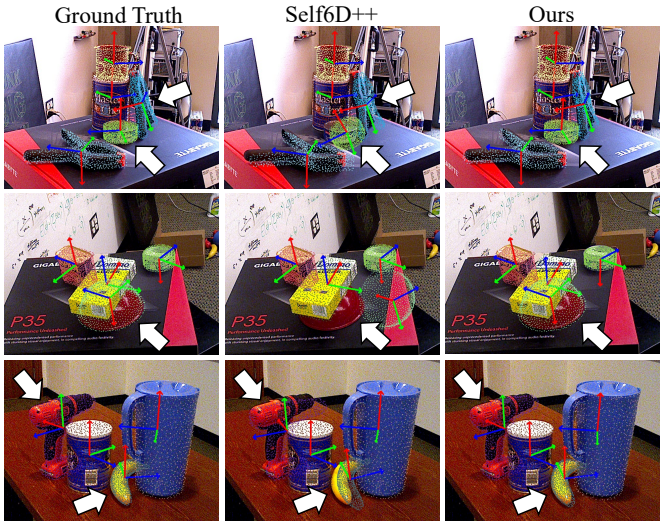
Models	$\mathcal{L}_{syn}$		$\mathcal{L}_{real}$		Object													Mean
	$\mathcal{L}_{rot+trans}$	$\mathcal{L}_{ort}$	$\mathcal{L}_{rec}$	$\mathcal{L}_{geo}$	Ape	Bvise	Cam	Can	Cat	Drill	Duck	Eggbox*	Glue*	Holep	Iron	Lamp	Phone	
$\mathcal{M}_1$				✓	21.9	26.5	2.4	5.1	13.6	17.9	1.6	23.0	38.4	0.0	25.4	1.3	18.7	15.1
$\mathcal{M}_2$			✓	✓	36.3	40.1	8.9	8.4	19.4	28.7	2.3	39.4	44.6	7.6	37.0	5.4	26.4	23.4
$\mathcal{M}_3$	✓				66.3	78.9	21.8	28.0	43.9	56.8	25.1	66.8	91.1	3.8	78.2	18.8	74.5	50.3
$\mathcal{M}_4$	✓	✓			63.4	74.8	29.0	20.1	37.9	61.2	34.7	79.6	94.2	4.9	65.5	21.3	82.0	51.4
$\mathcal{M}_5$	✓			✓	76.5	83.6	71.2	69.1	88.2	85.9	63.4	84.2	97.9	59.4	85.7	69.4	84.6	78.4
$\mathcal{M}_6$	✓	✓		✓	83.1	80.9	79.4	77.3	80.6	87.0	71.2	89.7	98.3	78.1	81.9	77.6	86.0	82.4
$\mathcal{M}_6^{\dagger}$	✓	✓		✓†	86.5	93.7	82.0	91.6	92.2	89.0	85.4	92.7	99.8	86.6	88.9	91.2	88.5	89.9
$\mathcal{M}_7$	✓		✓	✓	96.3	97.4	91.8	**94.6**	95.4	97.8	86.6	91.0	99.9	90.3	96.8	89.9	**93.1**	93.9
$\mathcal{M}_8$	✓	✓	✓	✓	**97.4**	**99.9**	**95.3**	94.4	**99.8**	**99.5**	**92.0**	**94.5**	**100.0**	**97.5**	**99.8**	**98.1**	92.3	**97.0**

TABLE V: Quantitative comparison of varying image patch processing strategies. Our method achieves the best performance with or without refinement (ICP) compared with the other two representative schemes, showing its effectiveness.

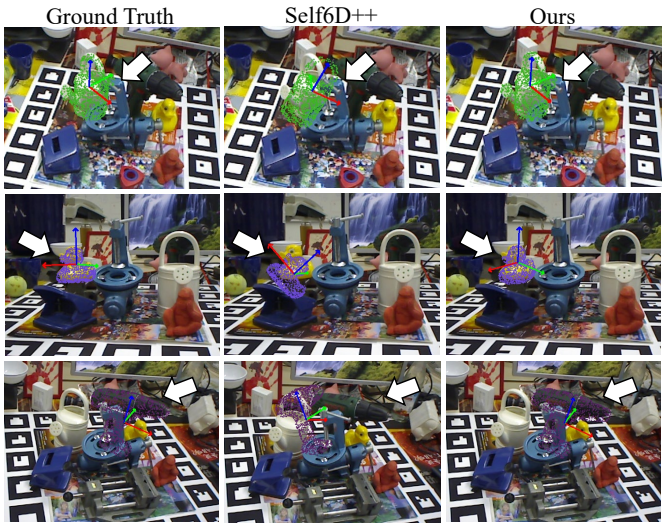
w/o ICP	W/o Resize	Resize Directly	Ours
Mean	81.9	88.6	**97.0**
w/ ICP	W/o Resize	Resize Directly	Ours
Mean	82.2	89.1	**97.3**

our method with other two conventional image patch process approaches, i.e., resizing the image patch to target dimension directly and without any resize procedure (keeping origin image patch dimension). As illustrated in Fig. 6, we report the ADD(S) less than 10% of the object's diameter (ADD(S)-0.1d) accuracy under the growth of the number of training iterations of six objects on the LineMOD test set. Compared with the other two approaches, our model can achieve more faster converged speed (20k-30k iterations) and reach higher pose accuracy. With the same number of iterations, our method outperforms the resize-directly approach by a large margin in general, demonstrating promising performance in training speed and pose accuracy. We think that is because our image patch process method is capable of maintaining the appearance structures of the target object in the cropped image, leading to better RGB feature extraction and benefit for subsequent pose estimation. Moreover, due to the limited training batchsize, the method without the image patch resize process shows severely slow training speed. Therefore, for fair comparisons, we also overlay the full ADD(S)-0.1d accuracy curve to explore its performance, i.e., w/o resize (full) in Fig. 6. It can be observed that our method surpasses it significantly both in converge speed and pose regression performance. In Tab. V, we also present the average ADD(S)-0.1d accuracy comparison results over the entire LineMOD dataset. Overall, our method significantly advances the other two approaches by 15.1% and 8.4% respectively. With the extra iterative refinement (e.g., ICP), our framework can make 0.3% accuracy improvements and still outperform all other methods by a large margin, demonstrating the effectiveness of our deployed image patch process algorithm.

Effect of components of the loss functions. To explore the impact of removing specific loss functions on the accuracy of pose estimation, we conducted comprehensive experiments on the LineMOD dataset. Table IV showcases the detailed pose results in terms of the ADD(S)-0.1d metric. Specifically, we first remove $\mathcal{L}_{syn}$ and investigate the individual component of $\mathcal{L}_{real}$. It has been observed that model $\mathcal{M}_1$ experiences significant degradation in pose accuracy when trained from scratch without any pose labels. Incorporating with extra shape completion loss $\mathcal{L}_{rec}$, model $\mathcal{M}_2$ makes 8.3% average accuracy improvements. We believe that having a good initialization for pose parameter regression is crucial for the network, similar to most optimization problems. Besides, due to the clutter scenario and challenging intra-object occlusions, the network struggles to optimize purely based on the geometry shape constraint, as the visible part of the object lacks consistency with the complete shape. We then remove $\mathcal{L}_{real}$ and train the network only on the synthetic data with the loss function $\mathcal{L}_{syn}$, and treat them as two baseline models (i.e., $\mathcal{M}_3$ and $\mathcal{M}_4$). We can find that due to the large domain gap between the synthetic and real-world data, these two baseline models achieve poor performance (i.e., 50.3% and 51.4% mean pose



(a) Qualitative comparison results on YCB-V benchmark.



(b) Qualitative comparison results on Occlusion LineMOD benchmark.

Fig. 5: Qualitative comparison of our method with self6D++ [1] on two benchmarks. Our method achieves superior performance in robustness to occlusion.

accuracy). Incorporating with the $\mathcal{L}_{geo}$, models $\mathcal{M}_5$ and $\mathcal{M}_6$ showcase significant performance improvements in pose accuracy, surpassing the baseline models by 28.1% and 31.0% respectively. These dramatic results indicate the effectiveness of our employed sim-real joint training strategy, benefiting the optimization process in the training phase. Furthermore, with the additional $\mathcal{L}_{rec}$, our model ($\mathcal{M}_7$ and $\mathcal{M}_8$) is capable of further performance improvements, advancing $\mathcal{M}_5$ and $\mathcal{M}_6$ by 15.5% and 14.6% respectively. We believe that using $\mathcal{L}_{rec}$ as a supervision signal for the shape reconstruction network branch during the training phase can lead to more precise structure recovery and improved geometry shape constraints. Moreover, compared with the baseline model $\mathcal{M}_4$, our final network achieves 45.6% accuracy improvements, further substantiating the superiority of the utilized $\mathcal{L}_{real}$ and the sim-real joint training scheme.

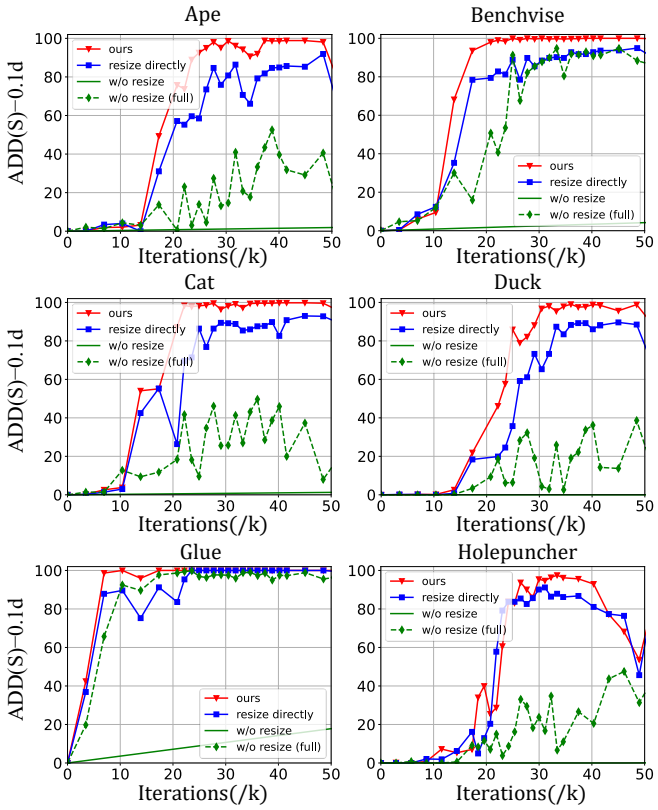


Fig. 6: Performance of different image patch processing strategies at increasing training iterations on LineMOD dataset. The ADD(S)-0.1d metric is reported. Our employed method performs effective performance both in accuracy and efficiency.

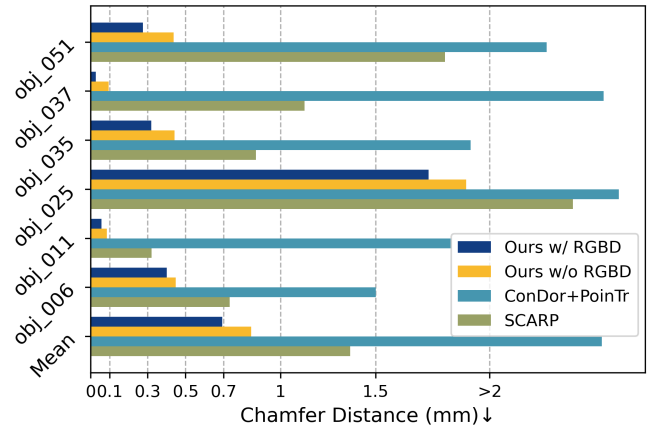


Fig. 7: Quantitative comparison of shape completion in arbitrary poses on YCB-V benchmark. The chamfer distance metric is reported and scaled by 10^3. Our method demonstrates better performance compared with the SOTA standard pipeline PoinTr [19] and the baseline model SCARP [30].

Effect of Canonical shape completion. To better investigate the effectiveness of the shape completion network, we conduct extensive experiments on the YCB-V dataset and compare it with other SOTA methods. As illustrated in Fig. 7, we report the chamfer distance of six objects derived from the YCB-V dataset and the overall average distance for

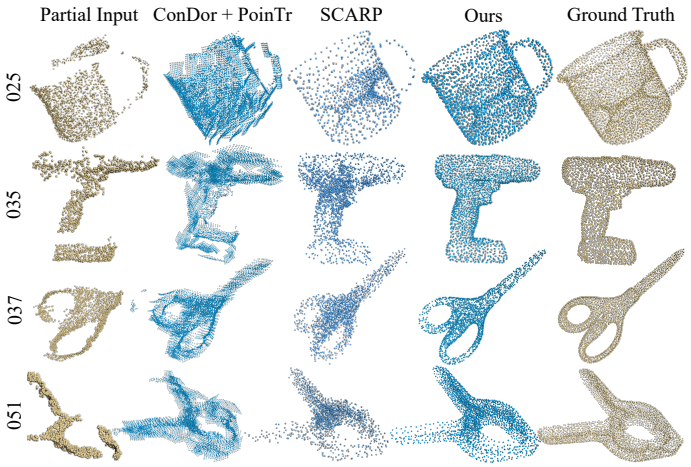


Fig. 8: Qualitative comparison of our method with the SOTA point cloud completion approach [19] and the baseline model [30] on the YCB-V benchmark. Our method produces more complete and detailed structures compared with its competitors, contributing to high pose accuracy results.

TABLE VI: Quantitative evaluation of segmentation results influence on YCB-V benchmark. We report the ADD(S) AUC and ADDS <2cm metrics. Note that the DF (Densefusion [12]) and Pr-GCN [21] are all trained with real pose labels.

Segmentation	PoseCNN		PVN3D		Ground Truth	
	ADD(S)	<2cm	ADD(S)	<2cm	ADD(S)	<2cm
DF(per-pixel)	91.2	95.3	91.5	95.7	92.9	96.8
Pr-GCN	95.0	97.6	95.8	98.5	96.9	99.9
Ours	91.6	96.0	92.3	96.8	93.7	98.6

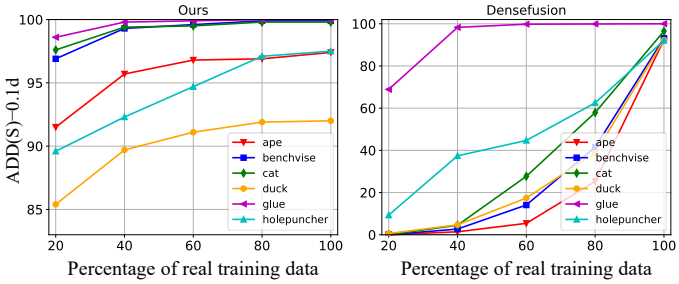


Fig. 9: Performance of different amounts of real training data on LineMOD dataset. The ADD(S)-0.1d metric is reported. Our weakly-supervised framework shows promising performance compared with the fully-supervised approach Densefusion [12].

entire object categories (21 categories) to evaluate the shape completion performance over the entire test set. We select two representative methods for comparison. SCARP [30], the baseline, is the first to use the TFN for shape completion in robotic grasping tasks, similar to our canonical shape reconstruction module. PoinTr [19] is the standard SOTA point cloud completion framework, utilizing transformer-based architectures. Specifically, we ablate the shape completion branch by removing the aggregated RGB-D features to explore its performance, then compare it with SCARP [30]. To ensure a fair comparison with the standard point cloud completion framework, we also follow [30] by utilizing the ConDor method [83] to canonicalize the observed partial point cloud to a fixed frame before integrating it into PoinTr [19]. As we can see, our model achieves better performance with the lowest chamfer distance over these six objects, whether 'with (w/)' or 'without (w/o)' fused RGBD features. By incorporating the fused RGBD features, our method can achieve the best performance compared with other approaches. Notably, in terms of the overall average chamfer distance, our model (w/ RGBD) achieves 49.3% performance improvements compared with the baseline model SCARP and outperforms the model (w/o RGBD) by 18.1% as well as large margin advancements compared with the PoinTr [19]. These experimental results demonstrate the effectiveness of the proposed shape reconstruction network, showing superiority in handling real-scanned shape completion problems with arbitrary poses. Moreover, we present the qualitative comparison results of four objects derived from the YCB-V dataset in Fig. 8. Note that the partial inputs are all taken from the camera coordinate system and mean-centered, which means that it is not in a unified frame with the ground truth shape. Compared with PoinTr [19] and SCARP [30], our method produces smoother surfaces and precise geometry structures. The visual experimental results reveal that the proposed method is capable of effectively recovering shape structures from the partial input with varying degrees of incompleteness, which plays an essential role in high-accuracy pose estimation (e.g., $\mathcal{M}_{7,8}$ in Tab. IV). To further validate the impact of pose accuracy promotion by the shape completion branch, we remove all the completion networks and take the CAD model as the supervision signal for $\mathcal{L}_{geo}$, i.e., model $\mathcal{M}_6^{\dagger}$ in Tab. IV. It is observed that by incorporating the shape completion branch, our models ($\mathcal{M}_{7,8}$) achieve 4.0% and 7.1% improvements respectively, substantiating its effectiveness. We believe that incorporating equivariant feature learning allows for a more comprehensive representation of an object's global geometric structure under arbitrary poses. This implicit representation enhances the multi-modal encoder in the pose estimation branch, effectively capturing global geometric features and compensating for inadequate feature representation due to incomplete observation data, thereby improving final pose accuracy.

The impact of segmentation results. As shown in Fig. 2, our framework takes the cropped foreground RGB-D images as input by using the off-the-shelf segmentation method. Therefore, to validate the influence of varying segmentation approaches, we follow Pr-GCN [21] to replace the segmentation method in our framework (i.e., PoseCNN [65], PVN3D [42] and the ground truth segmentation results) to evaluate the pose regression performance of our model. As illustrated in Tab. VI, we report the ADD(S) AUC and the ADD-S <2cm metrics tested on the YCB-V dataset, and compare our method with two fully-supervised approaches (i.e., Densefusion [12] and Pr-GCN [21]). It can be observed that by incorporating the ground truth results, all these methods achieve visible accuracy improvements in terms of these two metrics. In

TABLE VII: Quantitative evaluation in terms of ADD(S)-0.1d metric on the LineMOD dataset. *LB* and *UB* mean the lower-bound and upper-bound of our model. The *Ours*(*UB*) is obtained by fine-tuning our model with annotated real data.

Training data	Synthetic (PBR)		Syn+Real (w/ labels)		Syn+Real (w/o labels)
Methods	Self6D++ (LB) [1]	Ours (LB)	Self6D++ (UB) [1]	Ours (UB)	Ours (default)
ape	85.8	89.4	85.0	**98.3**	97.4
bvise	93.1	93.6	99.8	**100.0**	99.9
camera	**99.1**	93.4	96.5	97.9	95.3
can	**99.8**	94.1	99.3	98.7	94.4
cat	91.5	96.9	93.0	**99.9**	99.8
driller	**100.0**	99.3	**100.0**	99.7	99.5
duck	61.9	84.2	65.3	**94.6**	92.0
eggbox	93.5	94.0	**99.9**	97.9	94.5
glue	93.3	98.1	98.1	**100.0**	**100.0**
holep	32.1	87.6	73.4	**98.1**	97.5
iron	**100.0**	94.8	86.9	99.9	99.8
lamp	99.1	98.3	**99.6**	99.3	98.1
phone	**94.8**	91.8	86.3	94.7	92.3
MEAN	88.0	93.5	91.0	**98.4**	97.0

particular, our method outperforms Densefusion [12] in all three segmentation models and achieves comparable results compared with Pr-GCN [21]), demonstrating its effectiveness.

Performance regarding amount of real training data and pose labels. In Fig. 9, we evaluate our framework by utilizing varying quantities of real training data on six objects derived from LineMOD. As anticipated, the ADD(S) escalates in correlation with the expansion of real training data. Compared with the fully-supervised method [12], our model can achieve satisfactory accuracy even in a low level (20%) of real data usage (without any pose labels), demonstrating the effectiveness of our framework and low dependency on real pose labels. Moreover, a noteworthy enhancement in performance can be observed when real data usage is increased from 20% to 40%, after which it gradually ascends to its peak. This implies that our weakly-supervised framework can work well on small quantities of pose label-free real data, further substantiating its robustness. In addition, we present evaluation results of our framework using purely synthetic data, along with results obtained after fine-tuning with real pose labels, which serve as the lower and upper bounds for our method. We would like to highlight that our proposed framework can achieve better results in training with and without pose labels, compared to Self6D++ [1]. As shown in Tab. VII, our method showcases improvements of 5.5% and 7.4% in terms of the lower and upper bound on the average recall, respectively. By incorporating the proposed weakly-supervised strategy, our model advances the lower bound by 3.5% on average recall, approaching the upper bound. These experimental results demonstrate the effectiveness and superiority of our framework.

Performance with varying equivariance feature extractors. We choose the classic TFN network [56], [57] as our SE(3) equivariance feature learning backbone in the canonical shape reconstruction module, without loss of generality. To explore the effect of different SE(3) equivariance learning frameworks on the final pose estimation performance, we replace the TFN with the state-of-the-art point cloud equivariance representation learning networks, VNN [84] and E2PN [55]. The quantitative results are presented in Tab. VIII. As expected, the advanced equivariance learning model can further benefit the pose estimation performance of our framework. The VNN method shows significant performance improvements on two test objects, particularly in 'duck' category, with 4.3% accuracy increase. Similarly, the E2PN model achieves noticeable performance gains ranging from 0.1% to 3.8% on the four test objects, while maintaining comparable accuracy on remaining two objects when compared to our vanilla model. These experiments substantiate the effectiveness of SE(3) equivariance learning of the canonical shape reconstruction module, and the generalization ability of the proposed framework.

Performance with varying rotation representation types. In Tab. VIII, we conduct ablation studies on various rotation

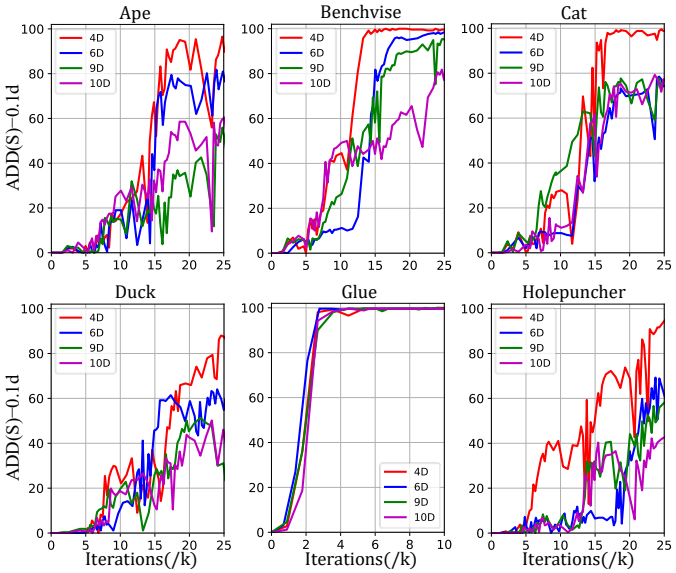


Fig. 10: Performance of different rotation representation methods at increasing training iterations on LineMOD dataset. The ADD(S)-0.1d metric is reported. The 4D rotation representation employed in our method performs better performance.

TABLE VIII: Quantitative evaluation results of varying rotation representation types and SE(3) equivariance feature extractors on LineMOD dataset. We report the ADD(S)-0.1d metric. † denotes our vanilla model.

Objs	Ape	Bvise	Cat	Duck	**Glue**	Holep	Params
	Varying se(3) equivariance feature extractors						
TFN†	97.4	99.9	99.8	92.0	**100.0**	**97.5**	**75.35**M
VNN [84]	**98.6**	99.4	99.7	**96.3**	**100.0**	96.9	77.11M
E2PN [55]	98.3	**100.0**	**100.0**	95.8	**100.0**	97.0	76.78M
	Varying rotation representation types						
4D†	**97.4**	**99.9**	**99.8**	**92.0**	**100.0**	97.5	**75.35**M
6D [85]	96.6	99.6	99.4	91.3	99.6	94.8	75.37M
9D [86]	97.1	99.8	99.7	91.0	**100.0**	**97.7**	75.39M
10D [87]	**97.4**	99.8	99.4	91.6	**100.0**	97.2	75.40M

TABLE IX: Runtime of Segmentation (Seg), Poes Regression (PR), Shape Reconstruction (SR) and the overall (Full). (Second per frame on LineMOD Dataset).

Component	Seg	PR	SR	Full
Times(s)	0.030	0.023	0.002	0.055

representation types using six objects derived from LineMOD. We replace the 4D output of our pose estimation branch with other rotation representation types in the form of continuous 6D [85], 9D [86], and 10D [87] vectors, respectively. The 4D representation demonstrates superior pose estimation performance across most object categories, while the 9D and 10D representations yield comparable but suboptimal results. In Fig. 10, we further present the ADD(S)-0.1d for these objects on the test set under increasing training iterations. It is evident that, due to the higher dimensions, the convergence speed of the model with the 6D, 9D, and 10D representations is slower than that of the 4D representation, requiring more iterations for the model to achieve optimal performance. These experiments demonstrate the compatibility of the 4D representation with our framework, allowing our model to achieve better results in both final pose estimation accuracy and convergence speed.

Time efficiency. In Tab. IX, we further follow [12] and [21] to evaluate the efficiency of our framework on LineMOD. Our full model achieves an acceptable runtime (55ms/frame), which is more efficient compared with Densefuion (60ms/frame) [12] as well as Pr-GCN (68ms/frame) [21] and can be satisfied for downstream robotic manipulation tasks [88].

V. CONCLUSION AND FUTURE WORK

In this work, we propose a novel weakly-supervised framework for instance-level object 6D pose estimation, in which we simultaneously consider shape completion in arbitrary poses and learning from the data without any real 6D pose annotations, addressing the problems of insufficient feature representation in incompleted depth data and the extensive demands for large amounts of real labeled training data. By incorporating the SE(3) equivariance, we can construct a fixed implicit frame to align the pose- and scale-independent partial inputs with the completed shape, thanks to the significant breakthroughs of SE(3) equivariance in 3D point cloud processing. Moreover, to bridge the domain gap, we employ a sim-real joint training strategy, making visible performance improvements. Extensive experiments on three widely used benchmarks demonstrate that our framework outperforms the existing state-of-the-art methods by a large margin, and significantly narrows the gap towards the SOTA approaches relying on real 6D pose annotations.

In the future, we will consider investigating extending our framework for category-level object pose estimation, as we currently rely on the 3D CAD model for each object. Another interesting aspect is the integration of the segmentation models into a unified end-to-end pipeline, by incorporating existing prevailing Big Models, suck as SAM [73].

REFERENCES

[1] G. Wang, F. Manhardt, X. Liu, X. Ji, and F. Tombari, "Occlusion-aware self-supervised monocular 6d object pose estimation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

[2] G. Zhou, D. Wang, Y. Yan, H. Chen, and Q. Chen, "Semi-supervised 6d object pose estimation without using real annotations," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 8, pp. 5163–5174, 2021.

[3] H.-S. Fang, C. Wang, M. Gou, and C. Lu, "Graspnet-1billion: A large-scale benchmark for general object grasping," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 11 444–11 453.

[4] M. Sundermeyer, A. Mousavian, R. Triebel, and D. Fox, "Contact-graspnet: Efficient 6-dof grasp generation in cluttered scenes," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 13 438–13 444.

[5] D. Xu, D. Anguelov, and A. Jain, "Pointfusion: Deep sensor fusion for 3d bounding box estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 244–253.

[6] Y. Nie, X. Han, S. Guo, Y. Zheng, J. Chang, and J. J. Zhang, "Total3dunderstanding: Joint layout, object pose and mesh reconstruction for indoor scenes from a single image," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 55–64.

[7] B. Drost, M. Ulrich, N. Navab, and S. Ilic, "Model globally, match locally: Efficient and robust 3d object recognition," in *2010 IEEE computer society conference on computer vision and pattern recognition*. Ieee, 2010, pp. 998–1005.

[8] S. Hinterstoisser, V. Lepetit, S. Ilic, S. Holzer, G. Bradski, K. Konolige, and N. Navab, "Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes," in *Computer Vision–ACCV 2012: 11th Asian Conference on Computer Vision, Daejeon, Korea, November 5-9, 2012, Revised Selected Papers, Part I 11*. Springer, 2013, pp. 548–562.

[9] J. Zhou, Y. Liu, J. Liu, Q. Xie, Y. Zhang, X. Zhu, and X. Ding, "Bold3d: A 3d bold descriptor for 6dof pose estimation," *Computers & Graphics*, vol. 89, pp. 94–104, 2020.

[10] K. Chen, S. James, C. Sui, Y.-H. Liu, P. Abbeel, and Q. Dou, "Stereopose: Category-level 6d transparent object pose estimation from stereo images via back-view nocs," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 2855–2861.

[11] J. Yang, Y. Gao, D. Li, and S. L. Waslander, "Robi: A multi-view dataset for reflective objects in robotic bin-picking," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 9788–9795.

[12] C. Wang, D. Xu, Y. Zhu, R. Martín-Martín, C. Lu, L. Fei-Fei, and S. Savarese, "Densefusion: 6d object pose estimation by iterative dense fusion," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3343–3352.

[13] Y. He, H. Huang, H. Fan, Q. Chen, and J. Sun, "Ffb6d: A full flow bidirectional fusion network for 6d pose estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 3003–3013.

[14] J. Zhou, K. Chen, L. Xu, Q. Dou, and J. Qin, "Deep fusion transformer network with weighted vector-wise keypoints voting for robust 6d object pose estimation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 13 967–13 977.

[15] J. Varley, C. DeChant, A. Richardson, J. Ruales, and P. Allen, "Shape completion enabled robotic grasping," in *2017 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 2017, pp. 2442–2447.

[16] D. Yang, T. Tosun, B. Eisner, V. Isler, and D. Lee, "Robotic grasping through combined image-based grasp proposal and 3d reconstruction," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 6350–6356.

[17] W. Gao and R. Tedrake, "kpam-sc: Generalizable manipulation planning using keypoint affordance and shape completion," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 6527–6533.

[18] W. Yuan, T. Khot, D. Held, C. Mertz, and M. Hebert, "Pcn: Point completion network," in *2018 international conference on 3D vision (3DV)*. IEEE, 2018, pp. 728–737.

[19] X. Yu, Y. Rao, Z. Wang, Z. Liu, J. Lu, and J. Zhou, "Pointr: Diverse point cloud completion with geometry-aware transformers," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 12 498–12 507.

[20] W. Zheng and Z. Han, "Snowflake point deconvolution for point cloud completion and generation with skip-transformer," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 5, 2023.

[21] G. Zhou, H. Wang, J. Chen, and D. Huang, "Pr-gcn: A deep graph convolutional network with point refinement for 6d pose estimation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 2793–2802.

[22] S. S. Mohammadi, N. F. Duarte, D. Dimou, Y. Wang, M. Taiana, P. Morerio, A. Dehban, P. Moreno, A. Bernardino, A. Del Bue *et al.*, "3dsgrasp: 3d shape-completion for robotic grasp," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 3815–3822.

[23] J. Gu, W.-C. Ma, S. Manivasagam, W. Zeng, Z. Wang, Y. Xiong, H. Su, and R. Urtasun, "Weakly-supervised 3d shape completion in the wild," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part V 16*. Springer, 2020, pp. 283–299.

[24] Y. Liu, K. Zhu, G. Wu, Y. Ren, B. Liu, Y. Liu, and J. Shan, "Mv-deepsdf: Implicit modeling with multi-sweep point clouds for 3d vehicle reconstruction in autonomous driving," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 8306–8316.

[25] E. Brachmann and C. Rother, "Visual camera re-localization from rgb and rgb-d images using dsac," *IEEE transactions on pattern analysis and machine intelligence*, vol. 44, no. 9, pp. 5847–5865, 2021.

[26] S. Peng, Y. Liu, Q. Huang, X. Zhou, and H. Bao, "Pvnet: Pixel-wise voting network for 6dof pose estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4561–4570.

[27] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*. Springer, 2014, pp. 740–755.

[28] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel, "Domain randomization for transferring deep neural networks from simulation to the real world," in *2017 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 2017, pp. 23–30.

[29] G. Wang, F. Manhardt, J. Shao, X. Ji, N. Navab, and F. Tombari, "Self6d: Self-supervised monocular 6d object pose estimation," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16*. Springer, 2020, pp. 108–125.

[30] B. Sen, A. Agarwal, G. Singh, B. Brojeshwar, S. Sridhar, and M. Krishna, "Scarp: 3d shape completion in arbitrary poses for improved grasping," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 3838–3845.

[31] W. Peng, J. Yan, H. Wen, and Y. Sun, "Self-supervised category-level 6d object pose estimation with deep implicit shape representation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 2, 2022, pp. 2082–2090.

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

[33] S. Hinterstoisser, S. Holzer, C. Cagniart, S. Ilic, K. Konolige, N. Navab, and V. Lepetit, "Multimodal templates for real-time detection of texture-less objects in heavily cluttered scenes," in *2011 international conference on computer vision*. IEEE, 2011, pp. 858–865.

[34] E. Brachmann, A. Krull, F. Michel, S. Gumhold, J. Shotton, and C. Rother, "Learning 6d object pose estimation using 3d object coordinates," in *European conference on computer vision*. Springer, 2014, pp. 536–551.

[35] K. Wada, E. Sucar, S. James, D. Lenton, and A. J. Davison, "Morefusion: Multi-object reasoning for 6d pose estimation from volumetric fusion," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 14 540–14 549.

[36] M. Tian, L. Pan, M. H. Ang, and G. H. Lee, "Robust 6d object pose estimation by learning rgb-d features," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 6218–6224.

[37] W. Chen, X. Jia, H. J. Chang, J. Duan, and A. Leonardis, "G2l-net: Global to local network for real-time 6d pose estimation with embedding vector features," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 4233–4242.

[38] G. Du, K. Wang, S. Lian, and K. Zhao, "Vision-based robotic grasping from object localization, object pose estimation to grasp estimation for parallel grippers: a review," *Artificial Intelligence Review*, vol. 54, no. 3, pp. 1677–1734, 2021.

[39] J. Liu, W. Sun, C. Liu, X. Zhang, S. Fan, and W. Wu, "Hff6d: Hierarchical feature fusion network for robust 6d object pose tracking," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 11, pp. 7719–7731, 2022.

[40] G. Feng, T.-B. Xu, F. Liu, M. Liu, and Z. Wei, "Nvr-net: Normal vector guided regression network for disentangled 6d pose estimation," *IEEE Transactions on Circuits and Systems for Video Technology*, 2023.

[41] G. Zhou, Y. Yan, D. Wang, and Q. Chen, "A novel depth and color feature fusion framework for 6d object pose estimation," *IEEE Transactions on Multimedia*, vol. 23, pp. 1630–1639, 2020.

[42] Y. He, W. Sun, H. Huang, J. Liu, H. Fan, and J. Sun, "Pvn3d: A deep point-wise 3d keypoints voting network for 6dof pose estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 11 632–11 641.

[43] T. Cao, W. Zhang, Y. Fu, S. Zheng, F. Luo, and C. Xiao, "Dgecn++: A depth-guided edge convolutional network for end-to-end 6d pose estimation via attention mechanism," *IEEE Transactions on Circuits and Systems for Video Technology*, 2023.

[44] Z. Yang, X. Yu, and Y. Yang, "Dsc-posenet: Learning 6dof object pose estimation via dual-scale consistency," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 3907–3916.

[45] X. Deng, Y. Xiang, A. Mousavian, C. Eppner, T. Bretl, and D. Fox, "Self-supervised 6d object pose estimation for robot manipulation," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 3665–3671.

[46] K. Chen, R. Cao, S. James, Y. Li, Y.-H. Liu, P. Abbeel, and Q. Dou, "Sim-to-real 6d object pose estimation via iterative self-training for robotic bin picking," in *European Conference on Computer Vision*. Springer, 2022, pp. 533–550.

[47] Y. Yang, C. Feng, Y. Shen, and D. Tian, "Foldingnet: Point cloud auto-encoder via deep grid deformation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 206–215.

[48] Z. Huang, Y. Yu, J. Xu, F. Ni, and X. Le, "Pf-net: Point fractal network for 3d point cloud completion," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 7662–7670.

[49] P. Xiang, X. Wen, Y.-S. Liu, Y.-P. Cao, P. Wan, W. Zheng, and Z. Han, "Snowflakenet: Point cloud completion by snowflake point deconvolution with skip-transformer," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 5499–5509.

[50] X. Wen, P. Xiang, Z. Han, Y.-P. Cao, P. Wan, W. Zheng, and Y.-S. Liu, "Pmp-net: Point cloud completion by learning multi-step point moving paths," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 7443–7452.

[51] H. Zhou, Y. Cao, W. Chu, J. Zhu, T. Lu, Y. Tai, and C. Wang, "Seedformer: Patch seeds based point cloud completion with upsample transformer," in *European conference on computer vision*. Springer, 2022, pp. 416–432.

[52] Z. Chen, F. Long, Z. Qiu, T. Yao, W. Zhou, J. Luo, and T. Mei, "Anchorformer: Point cloud completion from discriminative nodes," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 13 581–13 590.

[53] Z. Zhu, H. Chen, X. He, W. Wang, J. Qin, and M. Wei, "Svdformer: Complementing point cloud via self-view augmentation and self-structure dual-generator," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 14 508–14 518.

[54] H. Chen, S. Liu, W. Chen, H. Li, and R. Hill, "Equivariant point network for 3d point cloud analysis," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 14 514–14 523.

[55] M. Zhu, M. Ghaffari, W. A. Clark, and H. Peng, "E2pn: Efficient se (3)-equivariant point network," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 1223–1232.

[56] A. Poulenard and L. J. Guibas, "A functional approach to rotation equivariant non-linearities for tensor field networks." in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 13 174–13 183.

[57] N. Thomas, T. Smidt, S. Kearnes, L. Yang, L. Li, K. Kohlhoff, and P. Riley, "Tensor field networks: Rotation-and translation-equivariant neural networks for 3d point clouds," *arXiv preprint arXiv:1802.08219*, 2018.

[58] F. Fuchs, D. Worrall, V. Fischer, and M. Welling, "Se (3)-transformers: 3d roto-translation equivariant attention networks," *Advances in neural information processing systems*, vol. 33, pp. 1970–1981, 2020.

[59] M. Weiler, M. Geiger, M. Welling, W. Boomsma, and T. S. Cohen, "3d steerable cnns: Learning rotationally equivariant features in volumetric

data," *Advances in Neural Information Processing Systems*, vol. 31, 2018.
[60] D. W. Shu, S. W. Park, and J. Kwon, "3d point cloud generative adversarial network based on tree structured graph convolutions," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 3859–3868.
[61] P. Achlioptas, O. Diamanti, I. Mitliagkas, and L. Guibas, "Learning representations and generative models for 3d point clouds," in *International conference on machine learning*. PMLR, 2018, pp. 40–49.
[62] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2881–2890.
[63] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.
[64] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein, and J. M. Solomon, "Dynamic graph cnn for learning on point clouds," *ACM Transactions on Graphics (TOG)*, 2019.
[65] Y. Xiang, T. Schmidt, V. Narayanan, and D. Fox, "Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes," *arXiv preprint arXiv:1711.00199*, 2017.
[66] Y. Yan and G. S. Chirikjian, "Almost-uniform sampling of rotations for conformational searches in robotics and structural biology," in *2012 IEEE International Conference on Robotics and Automation*. IEEE, 2012, pp. 4254–4259.
[67] J. Diebel *et al.*, "Representing attitude: Euler angles, unit quaternions, and rotation vectors," *Matrix*, vol. 58, no. 15-16, pp. 1–35, 2006.
[68] X. Li, Y. Weng, L. Yi, L. J. Guibas, A. Abbott, S. Song, and H. Wang, "Leveraging se (3) equivariance for self-supervised category-level object pose estimation from point clouds," *Advances in neural information processing systems*, vol. 34, pp. 15 370–15 381, 2021.
[69] A. X. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su *et al.*, "Shapenet: An information-rich 3d model repository," *arXiv preprint arXiv:1512.03012*, 2015.
[70] K. Mo, S. Zhu, A. X. Chang, L. Yi, S. Tripathi, L. J. Guibas, and H. Su, "PartNet: A large-scale benchmark for fine-grained and hierarchical part-level 3D object understanding," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.
[71] T. Wu, L. Pan, J. Zhang, T. WANG, Z. Liu, and D. Lin, "Density-aware chamfer distance as a comprehensive metric for point cloud completion," in *In Advances in Neural Information Processing Systems (NeurIPS), 2021*, 2021.
[72] Y. Hu, J. Hugonot, P. Fua, and M. Salzmann, "Segmentation-driven 6d object pose estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3385–3394.
[73] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, P. Dollár, and R. Girshick, "Segment anything," *arXiv:2304.02643*, 2023.
[74] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.
[75] L. N. Smith, "Cyclical learning rates for training neural networks," in *2017 IEEE winter conference on applications of computer vision (WACV)*. IEEE, 2017, pp. 464–472.
[76] H. Lin, S. Peng, Z. Zhou, and X. Zhou, "Learning to estimate object poses without real image annotations," in *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, L. D. Raedt, Ed. International Joint Conferences on Artificial Intelligence Organization, 7 2022, pp. 1159–1165, main Track. [Online]. Available: https://doi.org/10.24963/ijcai.2022/162
[77] T. Tan and Q. Dong, "Smoc-net: Leveraging camera pose for self-supervised monocular object pose estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 21 307–21 316.
[78] H. Chen, F. Manhardt, N. Navab, and B. Busam, "Texpose: Neural texture learning for self-supervised 6d object pose estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 4841–4852.
[79] Y. Hai, R. Song, J. Li, D. Ferstl, and Y. Hu, "Pseudo flow consistency for self-supervised 6d object pose estimation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 14 075–14 085.
[80] T. Hodaň, M. Sundermeyer, B. Drost, Y. Labbé, E. Brachmann, F. Michel, C. Rother, and J. Matas, "Bop challenge 2020 on 6d object localization," in *Computer Vision–ECCV 2020 Workshops: Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*. Springer, 2020, pp. 577–594.
[81] S. Hinterstoisser, V. Lepetit, S. Ilic, S. Holzer, G. Bradski, K. Konolige, and N. Navab, "Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes," in *Asian conference on computer vision*. Springer, 2012, pp. 548–562.
[82] P. J. Besl and N. D. McKay, "Method for registration of 3-d shapes," in *Sensor fusion IV: control paradigms and data structures*, vol. 1611. Spie, 1992, pp. 586–606.
[83] R. Sajnani, A. Poulenard, J. Jain, R. Dua, L. J. Guibas, and S. Sridhar, "Condor: Self-supervised canonicalization of 3d pose for partial shapes," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 16 969–16 979.
[84] C. Deng, O. Litany, Y. Duan, A. Poulenard, A. Tagliasacchi, and L. J. Guibas, "Vector neurons: A general framework for so (3)-equivariant networks," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 12 200–12 209.
[85] Y. Zhou, C. Barnes, J. Lu, J. Yang, and H. Li, "On the continuity of rotation representations in neural networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 5745–5753.
[86] J. Levinson, C. Esteves, K. Chen, N. Snavely, A. Kanazawa, A. Rostamizadeh, and A. Makadia, "An analysis of svd for deep rotation estimation," *Advances in Neural Information Processing Systems*, vol. 33, pp. 22 554–22 565, 2020.
[87] V. Peretroukhin, M. Giamou, D. M. Rosen, W. N. Greene, N. Roy, and J. Kelly, "A Smooth Representation of SO(3) for Deep Rotation Learning with Uncertainty," in *Proceedings of Robotics: Science and Systems (RSS'20)*, Jul. 12–16 2020.
[88] D. Morrison, P. Corke, and J. Leitner, "Learning robust, real-time, reactive robotic grasping," *The International journal of robotics research*, vol. 39, no. 2-3, pp. 183–201, 2020.

Jun Zhou is currently pursuing the Ph.D. degree with the The Hong Kong Polytechnic University (POLYU), Hong Kong. He received the B.Eng. and M.Eng. degree in mechatronic engineering from Nanjing Agricultural University (NJAU) and Nanjing University of Aeronautics and Astronautics (NUAA), China, in 2018 and 2021 respectively. His research interests include object pose estimation, point cloud segmentation, and 2D/3D medical image registration.

Kai Chen (Student Member, IEEE) received the B.Eng. degree in remote sensing science and technique from Wuhan University, Wuhan, China, in 2016, and the M.Eng. degree in photogrammetry and remote sensing from the same university in 2019. He is currently working toward the Ph.D. degree in computer science and engineering, with The Chinese University of Hong Kong. His research interests include point cloud understanding, object pose estimation, and robot manipulation.

Mingqiang Wei (Senior Member, IEEE) received the Ph.D. degree in computer science and engineering from The Chinese University of Hong Kong (CUHK), Hong Kong, in 2014. He is currently a Professor with the School of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics (NUAA), Nanjing, China. Before joining NUAA, he was an Assistant Professor with the Hefei University of Technology and a Postdoctoral Fellow with CUHK. His research interests include 3D vision, computer graphics, and deep learning. He was a recipient of the CUHK Young Scholar Thesis Awards in 2014. He is also an Associate Editor of ACM TOMM, The Visual Computer, and Journal of Electronic Imaging, and the Guest Editor of IEEE TRANSACTIONS ON MULTIMEDIA. He has published more than 150 research papers, including IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, SIGGRAPH, IEEE TRANSACTIONS ON VISUALIZATION AND COMPUTER GRAPHICS, CVPR, and ICCV.

Xiao-Ping Zhang (Fellow, IEEE) received B.S. and Ph.D. degrees from Tsinghua University, in 1992 and 1996, respectively, both in Electronic Engineering. He holds an MBA in Finance, Economics and Entrepreneurship with Honors from the University of Chicago Booth School of Business, Chicago, IL.

He is Chair Professor at Tsinghua Shenzhen International Graduate School (SIGS) and Tsinghua-Berkeley Shenzhen Institute (TBSI), Tsinghua University. He was the founding Dean of Institute of Data and Information (iDI) at Tsinghua SIGS. He had been with the Department of Electrical, Computer and Biomedical Engineering, Toronto Metropolitan University (Formerly Ryerson University), Toronto, ON, Canada, as a Professor and the Director of the Communication and Signal Processing Applications Laboratory (CASPAL), and has served as the Program Director of Graduate Studies. His research interests include image and multimedia content analysis, sensor networks and IoT, machine learning/AI/robotics, statistical signal processing, and applications in big data, finance, and marketing.

Dr. Zhang is Fellow of the Canadian Academy of Engineering, Fellow of the Engineering Institute of Canada, Fellow of the IEEE, a registered Professional Engineer in Ontario, Canada, and a member of Beta Gamma Sigma Honor Society. He is the general Co-Chair for the IEEE International Conference on Acoustics, Speech, and Signal Processing, 2021. He is the general co-chair for 2017 GlobalSIP Symposium on Signal and Information Processing for Finance and Business, and the general co-chair for 2019 GlobalSIP Symposium on Signal, Information Processing and AI for Finance and Business. He was an elected Member of the ICME steering committee. He is the general chair for ICME2024 and BioCAS2023. He is Editor-in-Chief for the IEEE JOURNAL OF SELECTED TOPICS IN SIGNAL PROCESSING. He is Senior Area Editor for the IEEE TRANSACTIONS ON IMAGE PROCESSING. He served as Senior Area Editor the IEEE TRANSACTIONS ON SIGNAL PROCESSING and Associate Editor for the IEEE TRANSACTIONS ON IMAGE PROCESSING, the IEEE TRANSACTIONS ON MULTIMEDIA, the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, the IEEE TRANSACTIONS ON SIGNAL PROCESSING, and the IEEE SIGNAL PROCESSING LETTERS. He was selected as IEEE Distinguished Lecturer by the IEEE Signal Processing Society and by the IEEE Circuits and Systems Society.

Qi Dou (Member, IEEE) received the B.Eng. degree in biomedical engineering from Beihang University, Beijing, China, in 2014, and the Ph.D. degree in computer science with The Chinese University of Hong Kong (CUHK), Hong Kong, in 2018. She is currently an Assistant Professor with the Department of Computer Science and Engineering, (CUHK). She was a Postdoctoral Researcher with the Department of Computing, Imperial College London, London, U.K., from 2018 to 2020. Her research interests include interdisciplinary field of artificial intelligence and robotics for medical and industrial scenarios, with expertise in multimodal data analysis, and robot sensing and learning in dynamic environments.

Jing Qin (Senior Member, IEEE) is currently a Professor with the School of Nursing, The Hong Kong Polytechnic University, and a Key Member of the Centre for Smart Health. His research interests include creatively leveraging advanced virtual reality (VR) and artificial intelligence (AI) techniques in healthcare and medicine applications and his achievements in relevant areas have been well recognized by the academic community. He won the Hong Kong Medical and Health Device Industries Association Student Research Award for the Ph.D. study on VR-based simulation systems for surgical training and planning. He won three best paper awards for his research on AI-driven medical image analysis and computer-assisted surgery, including one of the most prestigious awards in this field: the MIA-MICCAI Best Paper Award in 2017. He served as the Local Organization Chair for MICCAI 2019, a technical program committee (TPC) member for many academic conferences, a speaker for many invited talks, and a referee for many prestigious journals in relevant fields.