Exploiting Point-Wise Attention in 6D Object Pose Estimation Based on Bidirectional Prediction

Yuhao Yang*, Jun Wu*, Yue Wang, Guangjian Zhang and Rong Xiong†

***Abstract*— Traditional geometric registration based estimation methods only exploit the CAD model implicitly, which leads to their dependence on observation quality and deficiency to occlusion.To address the problem,the paper proposes a bidirectional correspondence prediction network with a point-wise attention-aware mechanism. This network not only requires the model points to predict the correspondence but also explicitly models the geometric similarities between observations and the model prior. Our key insight is that the correlations between each model point and scene point provide essential information for learning point-pair matches. To further tackle the correlation noises brought by feature distribution divergence, we design a simple but effective pseudo-siamese network to improve feature homogeneity. Experimental results on the public datasets of LineMOD, YCB-Video, and Occ-LineMOD show that the proposed method achieves better performance than other state-of-the-art methods under the same evaluation criteria. Its robustness in estimating poses is greatly improved, especially in an environment with severe occlusions.**

I. INTRODUCTION

The object 6D pose estimation task is to compute the object's 3D rotation and 3D translation in the current scene with respect to the canonical coordinates. It is an essential problem in human-robot interaction applications such as augmented reality [1], autonomous driving [2], and robot manipulation [3]. Unlike category-level or unseen object pose estimation tasks [4], [5], When tackling the instance-level object pose estimation problem, a CAD model of the target object is generally specified. The model establishes the canonical coordinates, and contains the distinctive features of the target, providing vital prior for estimation. Herein lies one of the key research issues - how to utilize the CAD model for object pose estimation.

Some approaches [6], [7] intuitively exploit the CAD model by generating observations of the model under different poses with perspective projection, then comparing the query scene with the generated observations, and optimizing the pose to descend their differences. They directly harness the rendered features from the CAD model, but are rather sensitive to the initial guess and prone to a local minimum because the mapping between the pose and observation is not unimodal. On the other hand, some methods leverage the powerful fitting capability of the neural network to extract features for predicting the correspondence between the scene and the model [8], [9]. This pipeline adopts ground truth correspondences as supervision, obtaining global solutions to achieve higher accuracy. Though the CAD model is explicitly utilized in the registration stage, its point features are not fully exploited in the prediction process, causing dependence on the quality of observation. When the observations are incomplete or noisy, the global solution could also be affected. Recently, [10] attempts to further exploit the CAD model by introducing an inverse prediction process that predicts the corresponding scene points for each model point. But they only employ the averaged global features with point-wise features from each point set, disregarding their mutual attention that is obliging for correspondence prediction.

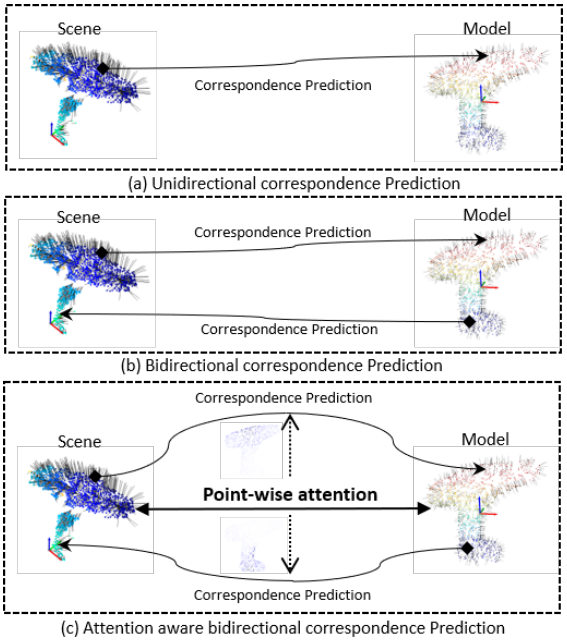

Fig. 1. **Illustration of our idea.** We show the difference between unidirectional match prediction methods (a), bidirectional match prediction methods (b), and our point-wise attention bidirectional match prediction method (c).

In this paper, we propose a bidirectional point-wise attention aware network for stable 6D object pose estimation. We adopt two branches to predict the correspondence from scene points to model coordinates and vice versa, and design a geometric attention mechanism to assist the prediction. Our key insight is that the correlations between each model point and scene point provide essential information for learning point-pair matches. The scene points are essentially

*These authors contributed equally to this work.

†Corresponding author rxiong@zju.edu.cn

Yuhao Yang and Guangjian Zhang are with the School of Artificial Intelligence, Chongqing University of Technology, Chongqing, 100190, China.

Jun Wu, Yue Wang, and Rong Xiong are with the College of Control science and Engineering, Zhejiang University, Hangzhou, 310027, China.

This work was supported in part by the National Nature Science Foundation of China under Grant 62173293 and in part by the Zhejiang Provincial Natural Science Foundation of China (LD22E050007).

model points observed in a specified view with noises and occlusions. Since the geometric properties of pointclouds do not vary with changes in viewing perspective, corresponding points in the scene and the model should still have the highest attention response to each other despite the transformation. During training, the known transformation is applied to supervise the learning of attention mechanism, enabling the network to model the point-wise attention during inference. This attention module is then concatenated with the feature vector to predict the correspondence. Experimental results demonstrate that by using the attention mechanism module, the correspondence prediction and pose estimation performance is improved.

We further leverage a simple yet effective pseudo-siamese network (PSN) to obtain point-wise attention. Intuitively, the coveted attention could be calculated directly from the features extracted for correspondence prediction [11]. However, we argue that these features are insufficient for attention awareness. The scene features often include color properties learned from the input RGB image, while the model features do not, which disturbs the mutual correlation computation. Moreover, the scene features and model features generally follow different distributions because they are extracted separately from two branches. We hope that the attention only reflects the geometric similarities between point pairs, rather than being affected by distribution divergences. Therefore, we design an additional pseudo-siamese neural network, which takes both sets of point clouds as input and extracts their features for attention calculation.

To summarize, the contributions of this paper are mainly as follows:

- To exploit the CAD model for stable 6D object pose estimation, we propose a bidirectional match prediction network with global point-wise attention aware mechanism, and prove its effectiveness in improving point pair match learning.
- To obtain robust attention, we introduce a simple but effective pseudo-siamese network to discover the similarities between model points and scene points.
- We validate our proposed method on public datasets of LineMOD, YCB-Video, and Occ-LineMOD. The experimental results show that our network outperforms state-of-the-art methods in both accuracy and robustness.

II. Related Works

A. Utilizing CAD models by comparison methods

Since the CAD model is available in the instance-level object pose estimation task, early methods directly generate templates by projecting the 3D model with various angles, and estimate the query pose by finding the most similar template image [12][13][14]. [6][15] propose a novel image representation by spreading image gradient orientations and representing the object with a limited set of templates. [16] verify the candidates by matching features in different modalities and associate the approximate poses with each detected template as the initial value for further optimization. These methods take advantage of explicitly comparing with the CAD model and are capable of handling textureless objects, but the discretized template generation process leads to less accuracy.

Recently, some methods adopt graphical rendering techniques to generate pseudo observations in continuous pose space [17][18][19][20]. [7] exploits a deep learning-based pose refinement network to refine the initial pose iteratively by minimizing the differences between the observed image and the rendered image. [21] proposes a pose refinement method using the standard differentiable rendering and learning the texture of a 3D model via contrastive loss. [22] utilizes differentiable Levenberg-Marquardt optimization to refine the pose by minimizing the distance between the input and rendered image representations. [23] identify the relative pose given the current observation and a synthetic image rendered from the previous estimates.

These methods have shown great performance by adopting render and compare refinement as a post-processing step, but they are time-consuming and sensitive to initial guess.

B. Utilizing CAD models by prediction methods

Instead of explicitly utilizing the generated images from the CAD model, another pipeline requires the network to implicitly learn the correlation between the observation and the model. With the recent advancements in deep learning, several methods [24][25][26][27] have attempted to detect box corners in the RGB image for 3D bounding box estimation. PVNet [9], employs farthest point sampling to vote for key points on the target object, predicting the direction vector pointing from each pixel to the projection point using a RANSAC voting strategy to locate the projection point. Some subsequent methods attempt to enhance the precision of correspondence prediction or the robustness of geometric solver [28][29][8][30][31]. Recently, some approaches have attempted to include model prior information to provide geometric constraints. [32] proposes a graph neural network to learn implicit neural representations of the 3D model and presents a dense correspondence matching scheme for visible points. BiCo-Net[10] adds an extra branch to predict the correspondence from model points to scene points, achieving higher robustness against occlusion. But they rely on the network to predict the correspondence correctly, ignoring the correlations between each point pair. While we further propose a global attention mechanism to leverage the difference among predicted correspondences.

Besides solving the registration problem from predicted correspondence, many other methods attempt to implicitly utilize the CAD model through direct regression approach [33][34][35]. DenseFusion [36] deploys a dense fusion strategy to fuse color features and geometry features point-wisely, then directly regresses pose from the fused feature. FFB6D [37] adds a bidirectional fusion module to fuse the two kinds of features on each encoding layer, bringing more local and global features. GCCN[11] applies a co-attention module to compute the correlations between scene points and model points. But they apply the attention

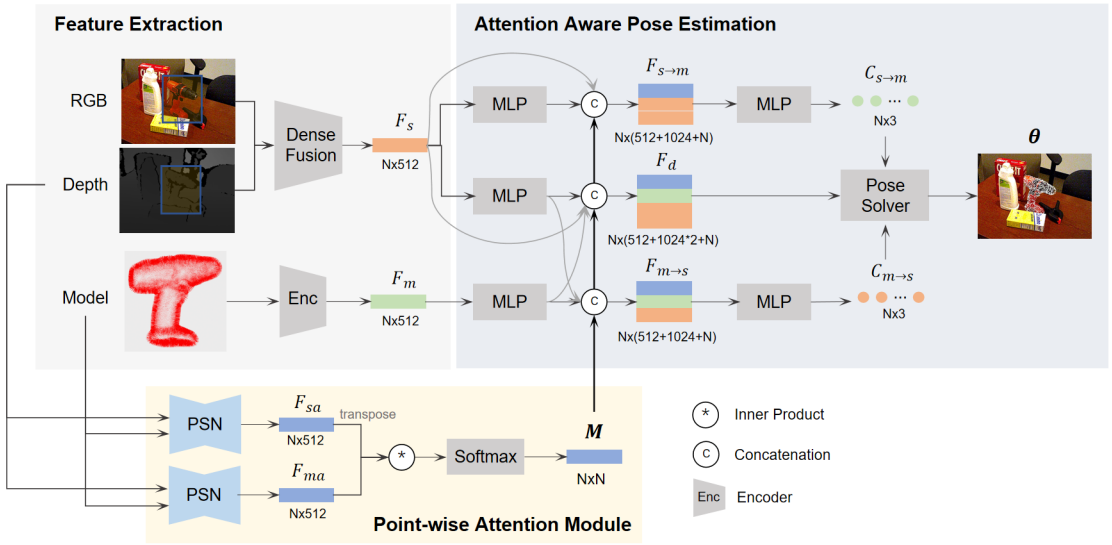

Fig. 2. Overview of our proposed method. We propose a point-wise attention module to obtain the correlations between scene points and model points. The attention is concatenated with other learned features to predict bidirectional correspondence and solve poses.

map with other features to directly regress the pose, reducing the influence of geometric properties. We show in the experiment section that our design of the pseudo-siamese network and geometric solver enhance the capability of the attention mechanism.

III. Methods

A. Overall Network Structure

The main structure of the network is illustrated in Fig. 2. Firstly, the interested region of the target object is cropped from the RGB and depth images as the input to the network. Then, we follow [10] to extract color and geometric features from the scene observations and fuse them point-wisely, as well as extract the model features. After feature extraction, we propose a point-wise attention module to model the correlations between the scene points and the model points. Last, we explain how we exploit the learned attention map for pose estimation.

B. Feature Extraction

Firstly, we crop the region of interest of the target object from the RGB and depth images as the input. Since segmentation is not the focus of our work, we use ground truth masks as in previous works [36].

To extract color features, we follow ConvNext [38] to extract surface texture features from the input RGB image. Then we randomly sample N points from the scene point cloud obtained from the depth image, and follow PointNet [39] to extract geometric features, which is further concatenated with their corresponding color embeddings to get the dense point-wise feature vector F_s. Besides, for bidirectional prediction, we also sample N points from the model points and extract features F_m from the model following PointNet [39].

C. Point-wise Attention Module

In this module, we consider the effect of global point pair geometric attention on the robustness of the final predicted poses and design the global point-wise attention module as shown in Fig. 2.

Pseudo-siamese Network. In order to obtain features for building the attention between scene points and model points, existing method [11] proposes to deploy two PointNets for each point set to extract their features, which are then compared to get their correlation. But these features follow different distributions. Therefore, we design a pseudo-siamese neural network, which takes both sets of point clouds as input. By doing so, the attention map only reflects the geometric similarities between point pairs, rather than being affected by distribution divergences.

As shown in Fig. 3, we input the sampled scene points and their normal vectors $(x, n) \in R^{N*6}$ into the network. Then 6 Conv1D layers are adopted to extract their geometric features. In order to preserve multi-level features, we perform short-circuits to connect the features from top layers to the last layer. Also, we observe that further concatenating exterior features from the model points could effectively advance the distribution consistency for afterward similarity computation. After concatenating the multi-level and exterior features, we obtain a fused feature of size $F \in R^{N*1048}$. Last, the fused feature is fed to an output Conv1D layer and then normalized to get the final scene point feature $F_{sa} \in R^{N*512}$. It is the same for the PSN to process model points to get $F_{ma} \in R^{N*512}$, except for that the exterior features are from scene points. Given F_{sa} and F_{ma}, we then apply inner production to take the two feature matrices as input and use the softmax function to generate the attention map M.

PPF constrains. To enhance the accuracy and effectiveness of the generated attention maps in focusing on geometric features, a PPF (Point Pair Feature) [12] constraint term is utilized as supervision to guide the attention maps. PPF is an effective way to calculate the relative positions and normal vector directions between point pairs, which enables it to capture the surface invariance and possess tolerance to pose

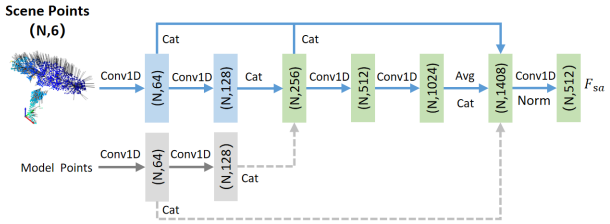

Fig. 3. Detailed structure of the proposed pseudo-siamese network.

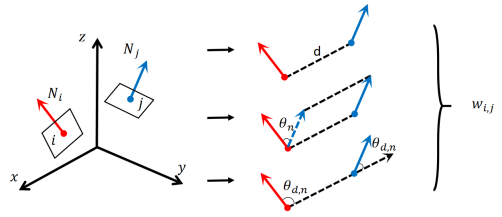

Fig. 4. Illustration of point-pair features for attention constraint.

changes. Additionally, the computation of PPF features takes into account the neighborhood information in the point cloud, which is beneficial in handling the case of partial occlusion. Specifically, all points in the scene point cloud are converted into the canonical coordinate using ground truth poses. Then each point in the transformed point cloud is compared with all points of the model point cloud to calculate the PPF-constrained point-pair features. As shown in Fig. 4, given the i_{th} transformed scene point $(x_i, n_i) \in \mathrm{R}^{N\times 6}$ and the j_{th} model point $(x_j, n_j) \in \mathrm{R}^{N\times 6}$, we calculate the Euclidean distance feature $d_{i,j}$, the normal vector angle feature $\theta_{i,j}$, and the distance vector and normal vector angle feature $\theta_{d_{i,j}}$ as follows

$$d_{i,j} = \|x_i - x_j\|_2 \tag{1}$$

$$\theta_{i,j} = \arccos\left(\frac{n_i \cdot n_j}{\|n_i\| \, \|n_j\|}\right) \tag{2}$$

$$\theta_{d_{i,j}} = \arccos\left(\frac{n_i}{\|n_i\|} \cdot \left(\frac{n_j}{\|n_j\|}\right)^T \cdot \frac{d_{i,j}}{\|d_{i,j}\|}\right) \tag{3}$$

After that, these three feature terms are weighted and aggregated as the final constraint term:

$$W(i,j) = \frac{1}{1 + (\gamma_1 d_{i,j} + \gamma_2 \theta_{d_{i,j}} + \gamma_3 \theta_{i,j})}, \tag{4}$$

where $\gamma_1, \gamma_2, \gamma_3$ are the weight parameters. During training, we supervise the learned attention map with this PPF constraint term

$$L_{attention} = \frac{1}{NN} \sum_{i=1}^{N} \sum_{j=1}^{N} (M(i,j) - W(i,j))^2 \tag{5}$$

D. Attention Aware Pose Estimation

Given the extracted features F_s and F_m, and the point-wise attention map M, we design an attention aware pose estimation mechanism. The key idea is to concatenate the attention map to the feature vectors to guide the correspondence matching and pose estimating process.

Specifically, we follow [10] to develop two different branches to separately predict the point matches $C_{s\to m}$ from the scene points to model points and the point matches $C_{m\to s}$ from the model points to scene points. In each branch, an MLP is deployed to decode the feature vector F_s or F_m, then the output features are concatenated with the attention map and fed to an MLP regressor for correspondence prediction.

In order to encode more features, we also deploy inter-branch concatenation and short circuits mechanism as shown in Fig. 2. Given predicted scene to model matches (x_i, n_i) and model to scene matches (x_j, n_j), we directly supervise the predicted correspondence with $L1$ losses

$$L_s = \frac{1}{N} \sum_i \left(\|x_i - \hat{x}_i\| + \varepsilon \left\|n_i^s - \hat{n}_i^s\right\|\right) \tag{6}$$

$$L_m = \frac{1}{N} \sum_j \left(\|x_j - \hat{x}_j\| + \varepsilon \left\|n_j^m - \hat{n}_j^m\right\|\right) \tag{7}$$

where ε is a hyper-parameter to balance the two terms.

Moreover, we also adopt another regression branch to directly predict the candidate poses $T_d = (R_d, t_d)$ from decoded features following [40] by regressing the 3D translation vector and a normalized 4D quaternion vector, in which the attention map is also concatenated in the way as in the other two branches. The poses are supervised with the ground truth pose with ADD loss for asymmetric objects

$$L_{d_i} = \frac{1}{K} \sum_k \left\|(R_{d_i} p_k + t_{d_i}) - \left(\hat{R}_{d_i} p_k + \hat{t}_{d_i}\right)\right\| \tag{8}$$

or with ADD-S loss for symmetric objects

$$L_{d_i} = \frac{1}{K} \sum_k \min_{j\in K} \left\|(R_{d_i} p_j + t_{d_i}) - \left(\hat{R}_{d_i} p_k + \hat{t}_{d_i}\right)\right\| \tag{9}$$

We train the network end-to-end with prediction losses, pose losses, and the attention loss together

$$L = \varphi_1 \frac{1}{N} \sum_i L_{d_i} + \varphi_2 L_s + \varphi_3 L_m + \varphi_4 L_{attention} \tag{10}$$

Last, we follow [12] to compute the possible poses T_s and T_m from the predicted point pairs. And due to the complementary nature of the information in these three pose sets, we merge the predicted poses from the three branches to obtain the final pose:

$$T_{final} = average(T_d \cup T_s \cup T_m) \tag{11}$$

IV. Experiment and Discussion

This section presents our experimental setup and implementation details and then reports the evaluation results on several commonly used datasets. We also demonstrate the effectiveness of our proposed components by performing several ablation studies.

TABLE I

EVALUATION RESULTS IN TERMS OF ADD(-S)(<0.1D) ON LINEMOD DATASET

	DenseFusion [36]	GCCN [11]	REDE [31]	G2L-Net [41]	PR-GCN [42]	PVN3D [8]	BiCo-net [10]	Ours
ape	92.3	97.5	95.6	96.8	97.6	97.3	97.3	**98.2**
benchvise	93.2	98.5	99.4	96.1	99.2	99.7	98.8	**99.7**
camera	94.4	99.7	99.6	98.2	99.4	99.6	99.6	**100.0**
can	93.1	99.5	99.5	98.0	98.4	99.5	99.3	**99.8**
cat	96.5	98.8	99.5	99.2	98.7	99.8	100.0	**100.0**
driller	87.0	96.6	99.3	**99.8**	98.8	99.3	98.9	99.3
duck	92.3	98.7	97.0	97.7	98.9	98.2	98.7	**99.0**
eggbox*	99.8	**100.0**	**100.0**	**100.0**	99.9	99.8	99.8	99.8
glue*	**100.0**	**100.0**	99.9	**100.0**	**100.0**	**100.0**	99.8	99.9
holepuncher	92.1	96.7	98.6	99.0	99.4	**99.9**	99.2	99.8
iron	97.0	97.1	99.3	99.3	98.5	99.7	**100.0**	99.9
lamp	95.3	99.1	99.3	99.5	99.2	99.8	99.7	**99.8**
phone	92.8	98.4	99.3	98.9	98.4	99.5	99.2	**99.5**
MEAN	94.3	98.5	98.9	98.7	98.9	99.4	99.3	**99.6**

* Objects marked with stars are symmetrical objects

A. Datasets

LineMOD [6] dataset contains a total of 13 objects, and we follow the approach in [40] to segment the training and testing data. Specifically, the dataset contains 13 low-texture objects placed in different cluttered environments, comprising 15783 images. And 1065 real data are randomly selected from the original dataset for testing.

YCB-Video [40] contains 21 shape and texture variations of YCB [43] objects. A subset of 92 RGBD videos of the objects is captured and annotated using 6D poses and instance semantic masks.

We follow [36] to use the GT mask for training, and divide the dataset into training and testing sets. 16189 frames plus 80,000 synthetic images provided by [40] are selected for training, and another 2949 critical frames from the remaining 12 videos are selected for testing.

Occ-LineMOD [44] dataset is a subset of the LineMOD dataset, containing 8 objects under severe occlusion and 1214 images with multiple severely occluded objects. We use this dataset to test the robustness of pose estimation in challenging situations.

B. Evalution Metrics

We used ADD [6] and ADD-S [40] evaluation metrics used by most methods to evaluate our model. ADD is the average Euclidean distance between the model points after transforming the predicted and ground truth poses. ADD-S is a metric for symmetric objects to calculate the average distance to the nearest point. In both LineMOD and Occ-LineMOD datasets, we report the accuracy of pose prediction for ADD(-S) $<$ 0.1d. While for the YCB-Video dataset, we report the area under the curve obtained by ADD(-S) by varying the distance threshold and the percentage of all ADD(-S) data less than 2 cm.

C. Experimental results

LineMOD. We evaluate our performance in the LineMOD dataset as shown in Table I. For a fair comparison, apart from [8] which predicts masks by their own, all other methods including ours utilize the masks provided by PoseCNN [40]. Our method uses only real data for training and outperforms all other methods, with a higher accuracy of more than 0.2%. For small objects in the dataset, it is demanding for other networks to estimate their poses effectively based on a small number of pixel points. Our method tightly links the model point cloud and depth information. Eventually, the prediction robustness of these objects is significantly improved compared to other methods. The performance of our approach exhibits a slight deficiency when applied to the symmetrical objects. We argue that it is attributed to the existence of multimodal responses in the learned attention maps that represent the correlation between model points and scene points, pertaining to the geometric symmetry of the objects. Consequently, these multimodal responses marginally impact the learning of the matching process.

YCB-Video. We evaluate our performance with the GT mask and the PVN3D[8] masks respectively for a fair comparison, as shown in Table II. Our method has an advantage over most of the state-of-the-art methods and achieves 99.1% on ADD-S (<2cm). Our accuracy improves on most objects thanks to the point-pair feature constraint. Fig. 5 shows the results of the visualization of the predicted poses of some of the objects. It can be seen that our method has improved the robustness of the network in predicting poses to some extent, and is able to accurately calculate the correct poses even with interference such as occlusion.

Occ-LineMOD. For the most challenging dataset, the final results are shown in Table III. It can be seen that our method shows a significant improvement in ADD-S<0.1d compared to other methods, with an average prediction accuracy of 74.4%. Notably, our method demonstrates improved performance on symmetrical objects in more occluded situations. We attribute this observation to the reduction in potential correlations between point pairs brought about by surface occlusions. This reduction in the likelihood of multimodal responses subsequently enhances the precision of the matching process. However, since the structure of PR-GCN [42] based on graph convolutional network can make full use of the geometric information and topology of objects in images, they also show better performance at handling topological information of objects.

TABLE II

EVALUATION RESULTS IN TERMS OF THE ADD-S(AUC) AND ADD-S(<2CM) EVALUATION METRICS ON YCB-VIDEO DATASET

	with GT mask						with PVN3D mask							
	DenseFusion [36]		BiCo-net [10]		Ours		PVN3D [8]		PR-GCN [42]		BiCo-net [10]		Ours	
Object	AUC	<2cm	AUC	<2cm	AUC	<2cm	AUC	<2cm	AUC	<2cm	AUC	<2cm	AUC	<2cm
002	96.2	100.0	96.3	100	**96.9**	**100**	96.0	100	**97.1**	100	96.4	100.0	95.8	**100**
003	95.3	100	96.5	100	**96.2**	**100**	96.1	100	**97.6**	100	96.1	99.9	96.5	**100**
004	97.9	100	97.5	100	**98**	**100**	97.4	100	**98.3**	100	97.9	100	98.1	**100**
005	94.3	96.9	96.4	98.7	**96.8**	**98.7**	**96.2**	98.1	95.3	97.6	95.8	**98.1**	95.9	98
006	97.7	100	98.0	100	**98**	**100**	97.5	100	97.9	100	97.9	100	**98.3**	**100**
007	**96.7**	100	95.9	100	96.5	**100**	96	100	**97.6**	100	96.2	100	97.1	**100**
008	97.3	100	97.7	100	**97.8**	**100**	97.1	100	**98.4**	100	97.3	100	98.1	**100**
009	98.4	100	98.3	100	**98.9**	**100**	97.7	100	96.2	94.4	**98.9**	100	98.7	**100**
010	90.2	92.3	93.1	**95.6**	**93.6**	95.4	93.3	94.6	**96.6**	**99.1**	93	94.7	93.4	94.8
011	96.2	99.7	97.4	100	**97.7**	**100**	96.6	100	**98.5**	100	97.4	100	97.6	**100**
019	97.5	100	97.0	100	**97.5**	**100**	97.4	100	**98.1**	100	97.5	100	97.6	**100**
021	96.4	100	97.0	100	**97.1**	**100**	96	100	**97.9**	100	96.4	100	96.8	**100**
024*	88.9	87.4	96.5	100	**97.0**	**100**	90.2	80.5	90.3	96.6	**96.5**	**100**	96	99.3
025	97.0	100	96.5	100	**97.3**	**100**	97.6	100	**98.1**	100	97.2	100	97.4	**100**
035	97.1	100	96.8	100	**97.1**	**100**	96.7	100	**98.1**	100	96.9	100	97.3	**100**
036*	94.1	100	95.2	100	**95.2**	**100**	90.4	93.8	**96**	**100**	91.5	89.7	94.1	98.8
037	93.2	100	**95**	100	94.5	**100**	**96.7**	100	**96.7**	100	90.8	98.9	93.5	**100**
040	97.5	100	**97.3**	100	97.2	**100**	96.7	99.8	97.9	100	96.8	100	**98**	**100**
051*	89.7	98.0	95.9	100	**95.9**	**100**	93.6	93.6	87.5	93.3	94.4	98.5	92	**98.5**
052*	77.4	80.5	95.1	99.9	**95.6**	**100**	88.4	83.6	79.7	84.6	88.4	91.2	86.9	**91.5**
061*	91.5	100	96.8	100	**97.3**	**100**	96.8	100	**97.8**	100	97.2	100	96.9	**100**
ALL	94.2	97.8	96.4	99.6	**96.7**	**96.6**	95.5	97.6	95.8	98.5	95.8	98.8	**95.8**	**99.1**

* Objects marked with stars are symmetrical objects

TABLE III

EVALUATION RESULTS IN TERMS OF ADD(-S)(<0.1D) ON OCC-LINEMOD DATASET

	PVNet [9]	REDE [31]	FFB6D [37]	PR-GCN [42]	BiCo-net [10]	Ours
ape	15.8	53.1	47.2	40.2	55.6	**58.3**
can	63.3	88.5	85.2	76.2	83.2	**88.5**
cat	16.7	35.9	45.7	**57.0**	47.3	51.6
driller	65.7	77.8	81.4	**82.3**	69.9	77.8
duck	25.2	46.2	53.9	30.0	58.3	**64.8**
eggbox*	50.2	71.8	70.2	68.2	78.1	**81.3**
glue*	49.6	75.0	60.1	67.0	76.9	**79.0**
holepuncher	39.7	75.5	85.9	**97.2**	87.2	93.6
Mean	40.8	65.4	66.2	65.0	69.5	**74.4**

* Objects marked with stars are symmetrical objects

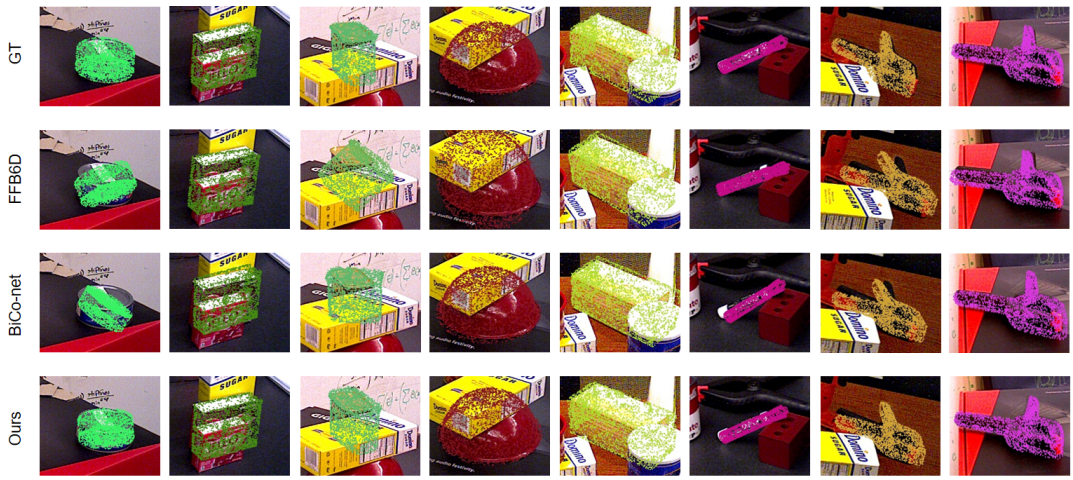

Fig. 5. Illustration of the performance of our method compared with other baseline methods on the YCB-Video dataset. Point clouds are projected back to the image after being transformed by the predicted pose. Images are cropped for better visualization.

D. Ablation Studies

Effect of attention aware pose estimation. To verify the impact of the point-wise attention map module on the robustness of the predicted poses, we conduct ablation studies on the LineMOD and Occ-LineMOD datasets. As shown in Table IV, we find that concatenating the attention map to the 3 branches makes them aware of the correlation between model points and scene observations, which significantly improves the subsequent pose prediction results.

Effect of supervising the attention map with PPF features. To verify the effect of the point-pair feature weights in supervising the point-wise attention map, we recombine

TABLE IV

EFFECT OF ATTENTION AWARE POSE ESTIMATION.

Attention for direct regression	Attention for match prediction	LineMOD ADD(-S)	Occ-LineMOD ADD(-S)
		99.3	69.5
	✓	99.4	72.4
✓		99.5	73.7
✓	✓	99.6	74.4

these three features and conduct ablation studies on the LineMOD and Occ-LineMOD datasets. As shown in Table V, if the PPF constraints term only uses the point distance vector feature and the angle of the normal feature, it ends up with 73.3% of the final results, which is a 3.8% improvement compared to the original network without using constrained weights. However, we add the angle between the normal and the distance vector as another feature constraint to jointly guide the global attention map, and the final result is improved by 1.1%.

TABLE V

EFFECT OF PPF WEIGHT CONSTRAINT TERMS

d	$\theta_{d,N}$	θ_N	LineMOD ADD(-S)	Occ-LineMOD ADD(-S)
✓			99.5	73.4
	✓		99.4	73.5
		✓	99.4	73.9
✓		✓	99.4	73.3
✓	✓		99.5	73.5
	✓	✓	99.4	74.0
✓	✓	✓	99.6	74.4

Effect of point-wise attention mechanism compared with GCCN. In order to validate the effectiveness of our proposed point-wise attention mechanism compared with GCCN [11], we conduct a series of ablation experiments on all the three datasets. As shown in Table VI, we first replace the pseudo-siamese network (PSN) to the feature extraction networks in GCCN. The experimental results show a significant decrease in the pose prediction performance compared to using PSN. Then, we apply the improved PPF weight constraint terms and achieve more improvements. Finally, as shown in Fig. 6, we visually compare the results by taking the point clouds from different viewpoints of the "can" object in the Occ-LineMOD dataset. It can be observed that our point-wise attention map can better model the weight distribution that reflects the correlation between the model and the scene, leading to higher matching precision.

TABLE VI

EFFECT OF PSEUDO-SIAMESE NETWORK (PSN)

PSN	PPF constraints	LineMOD (ADD-S)	Occ-LineMOD (ADD-S)	YCB-Video (<2cm)
		99.3	73.3	98.8
	✓	99.4	73.7	98.9
✓	✓	99.6	74.4	99.1

V. CONCLUSION

In this paper, we propose a bidirectional correspondence prediction network with point-wise attention aware mechanism to utilize a CAD model for stable 6D object pose estimation. Also, we introduce pseudo-siamese network to discover the similarities between model points and scene points, obtaining robust attention correlations. Experiments display that our method shows advantages over current state-of-the-art methods, and the accuracy and robustness of our prediction results are improved for 6D pose estimation. However, our performance still relies on the quality of the mask to a certain extent, which we have not addressed in this paper. We will consider this issue and try to implement associative segmentation and pose estimation in future works.

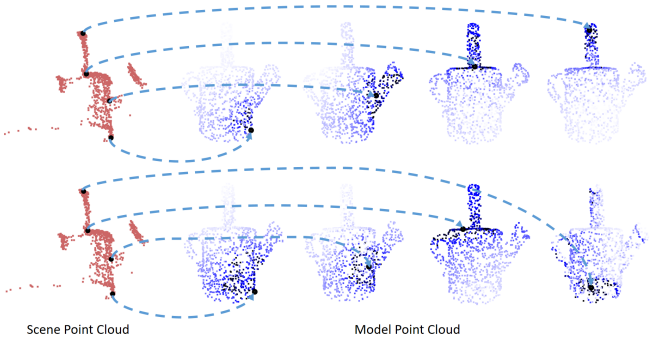

Fig. 6. Visualization of attention maps learned by our attention module (first row) and by GCCN method (second row). We select a point in the scene point cloud, and the corresponding attention map on the model point cloud is projected into a 2D image plane for visualization. The dotted lines connect the scene point and the corresponding points with the largest attention values.

REFERENCES

[1] Eric Marchand, Hideaki Uchiyama, and Fabien Spindler. Pose estimation for augmented reality: a hands-on survey. *IEEE transactions on visualization and computer graphics*, 22(12):2633–2651, 2015.

[2] Xiaozhi Chen, Huimin Ma, Ji Wan, Bo Li, and Tian Xia. Multi-view 3d object detection network for autonomous driving. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 1907–1915, 2017.

[3] Menglong Zhu, Konstantinos G Derpanis, Yinfei Yang, Samarth Brahmbhatt, Mabel Zhang, Cody Phillips, Matthieu Lecce, and Kostas Daniilidis. Single image 3d object detection and pose estimation for grasping. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3936–3943. IEEE, 2014.

[4] He Wang, Srinath Sridhar, Jingwei Huang, Julien Valentin, Shuran Song, and Leonidas J Guibas. Normalized object coordinate space for category-level 6d object pose and size estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2642–2651, 2019.

[5] Xu Chen, Zijian Dong, Jie Song, Andreas Geiger, and Otmar Hilliges. Category level object pose estimation via neural analysis-by-synthesis. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVI 16*, pages 139–156. Springer, 2020.

[6] Stefan Hinterstoisser, Stefan Holzer, Cedric Cagniart, Slobodan Ilic, Kurt Konolige, Nassir Navab, and Vincent Lepetit. Multimodal templates for real-time detection of texture-less objects in heavily cluttered scenes. In *2011 international conference on computer vision*, pages 858–865. IEEE, 2011.

[7] Yi Li, Gu Wang, Xiangyang Ji, Yu Xiang, and Dieter Fox. Deepim: Deep iterative matching for 6d pose estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 683–698, 2018.

[8] Yisheng He, Yimeng Sun, Jiashuo Huang, Xu Liu, Chuyang Fan, and Baoquan Li. 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*, pages 5499–5508, 2020.

[9] Sida Peng, Yuan Liu, Qixing Huang, Xiaowei Zhou, and Hujun Bao. Pvnet: Pixel-wise voting network for 6dof pose estimation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4561–4570, 2019.

[10] Zelin Xu, Yichen Zhang, Ke Chen, and Kui Jia. Bico-net: Regress globally, match locally for robust 6d pose estimation. *arXiv preprint arXiv:2205.03536*, 2022.

[11] Yongming Wen, Yiquan Fang, Junhao Cai, Kimwa Tung, and Hui Cheng. Gccn: Geometric constraint co-attention network for 6d object pose estimation. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 2671–2679, 2021.

[12] Bertram Drost, Markus Ulrich, Nassir Navab, and Slobodan Ilic. Model globally, match locally: Efficient and robust 3d object recognition. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 998–1005, 2010.

[13] Albert Gordo, Jon Almazán, Jerome Revaud, and Diane Larlus. Deep image retrieval: Learning global representations for image search. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VI 14*, pages 241–257. Springer, 2016.

[14] Chunhui Gu and Xiaofeng Ren. Discriminative mixture-of-templates for viewpoint classification. In *Computer Vision–ECCV 2010: 11th European Conference on Computer Vision, Heraklion, Crete, Greece, September 5-11, 2010, Proceedings, Part V 11*, pages 408–421. Springer, 2010.

[15] Stefan Hinterstoisser, Vincent Lepetit, Slobodan Ilic, Stefan Holzer, Gary Bradski, Kurt Konolige, and Nassir 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*, pages 548–562. Springer, 2013.

[16] Tomáš Hodaň, Xenophon Zabulis, Manolis Lourakis, Štěpán Obdržálek, and Jiří Matas. Detection and fine 3d pose estimation of texture-less objects in rgb-d images. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4421–4428. IEEE, 2015.

[17] Fabian Manhardt, Wadim Kehl, Nassir Navab, and Federico Tombari. Deep model-based 6d pose refinement in rgb. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 800–815, 2018.

[18] Ameni Trabelsi, Mohamed Chaabane, Nathaniel Blanchard, and Ross Beveridge. A pose proposal and refinement network for better 6d object pose estimation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2382–2391, 2021.

[19] Lin Yen-Chen, Pete Florence, Jonathan T Barron, Alberto Rodriguez, Phillip Isola, and Tsung-Yi Lin. inerf: Inverting neural radiance fields for pose estimation. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1323–1330. IEEE, 2021.

[20] Sergey Zakharov, Ivan Shugurov, and Slobodan Ilic. Dpod: 6d pose object detector and refiner. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1941–1950, 2019.

[21] Angtian Wang, Adam Kortylewski, and Alan Yuille. Nemo: Neural mesh models of contrastive features for robust 3d pose estimation. *arXiv preprint arXiv:2101.12378*, 2021.

[22] Shun Iwase, Xingyu Liu, Rawal Khirodkar, Rio Yokota, and Kris M Kitani. Repose: Fast 6d object pose refinement via deep texture rendering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3303–3312, 2021.

[23] Bowen Wen, Chaitanya Mitash, Baozhang Ren, and Kostas E Bekris. se(3)-tracknet: Data-driven 6d pose tracking by calibrating image residuals in synthetic domains. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 10367–10373. IEEE, 2020.

[24] Mahdi Rad and Vincent Lepetit. Bb8: A scalable, accurate, robust to partial occlusion method for predicting the 3d poses of challenging objects without using depth. In *Proceedings of the IEEE international conference on computer vision*, pages 3828–3836, 2017.

[25] Bugra Tekin, Sudipta N Sinha, and Pascal Fua. Real-time seamless single shot 6d object pose prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 292–301, 2018.

[26] Qianhui Luo, Huifang Ma, Li Tang, Yue Wang, and Rong Xiong. 3d-ssd: Learning hierarchical features from rgb-d images for amodal 3d object detection. *Neurocomputing*, 378:364–374, 2020.

[27] Chaitanya Mitash, Abdeslam Boularias, and Kostas Bekris. Robust 6d object pose estimation with stochastic congruent sets. *arXiv preprint arXiv:1805.06324*, 2018.

[28] Bowen Wen, Chaitanya Mitash, Sruthi Soorian, Andrew Kimmel, Avishai Sintov, and Kostas E Bekris. Robust, occlusion-aware pose estimation for objects grasped by adaptive hands. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6210–6217. IEEE, 2020.

[29] Eric Brachmann, Carsten Rother, Jens Konrad, and Ben Glocker. 6dof object pose estimation via differentiable proxy voting loss. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 7509–7518, 2019.

[30] Hansheng Chen, Pichao Wang, Fan Wang, Wei Tian, Lu Xiong, and Hao Li. Epro-pnp: Generalized end-to-end probabilistic perspective-n-points for monocular object pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2781–2790, 2022.

[31] Weitong Hua, Zhongxiang Zhou, Jun Wu, Huang Huang, Yue Wang, and Rong Xiong. Rede: End-to-end object 6d pose robust estimation using differentiable outliers elimination. *IEEE Robotics and Automation Letters*, 6(2):2886–2893, 2021.

[32] Chenrui Wu, Long Chen, Shenglong Wang, Han Yang, and Junjie Jiang. Geometric-aware dense matching network for 6d pose estimation of objects from rgb-d images. *Pattern Recognition*, page 109293, 2023.

[33] Gu Wang, Fabian Manhardt, Federico Tombari, and Xiangyang Ji. Gdr-net: Geometry-guided direct regression network for monocular 6d object pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16611–16621, 2021.

[34] Yan Di, Fabian Manhardt, Gu Wang, Xiangyang Ji, Nassir Navab, and Federico Tombari. So-pose: Exploiting self-occlusion for direct 6d pose estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12396–12405, 2021.

[35] Yinlin Hu, Pascal Fua, Wei Wang, and Mathieu Salzmann. Single-stage 6d object pose estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2930–2939, 2020.

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

[37] Yisheng He, Zhe Wu, Jun Wang, Ye Yuan, Zixiang Dong, and Wei Liu. Ffb6d: A full flow bidirectional fusion network for 6d pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5246–5255, 2021.

[38] Zhou Liu, Hongwei Mao, Chuangyu Wu, Yiming Sun, Rongrong Ji, and Zhiquan Su. A convnet for the 2020s. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11976–11986, 2022.

[39] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 652–660, 2017.

[40] Yu Xiang, Tanner Schmidt, Venkatraman Narayanan, and Dieter Fox. Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes. *arXiv preprint arXiv:1711.00199*, 2017.

[41] Wei Chen, Xiao Jia, Hyung Jin Chang, and et al. 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*, pages 4233–4242, 2020.

[42] Guangliang Zhou, Hao Wang, Qijun Chen, Yi Yan, and Deming Wang. 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*, pages 2793–2802, 2021.

[43] Berk Calli, Arpit Singh, Ariel Walsman, Siddhartha Srinivasa, Pieter Abbeel, and Aaron M Dollar. The ycb object and model set: Towards common benchmarks for manipulation research. In *2015 International Conference on Advanced Robotics (ICAR)*, pages 510–517. IEEE, 2015.

[44] Eric Brachmann, Alexander Krull, Frank Michel, Stefan Gumhold, Jamie Shotton, and Carsten Rother. Learning 6d object pose estimation using 3d object coordinates. In *European conference on computer vision*, pages 536–551. Springer, 2014.