FAST 6D OBJECT POSE REFINEMENT VIA IMPLICIT SURFACE REPRESENTATION DRIVEN OPTIMIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Pose refinement after the initial pose estimator has been demonstrated to be effective for 6D object pose estimation. The iterative closest point (ICP) is the most popular refinement strategy, which however suffers from slow convergence due to the nature of iterative nonlinear optimization. In this paper, we propose a simple yet efficient self-supervised point cloud aligenment method via implicit neural network, which can serve as an alternative of ICP to achieve fast and accurate pose refinement. Our key idea is to encode the surface of target point cloud into a signed distance function (SDF); the optimal rigid transformation then can be derived by addressing a minimization problem over the SDF. The workflow of our method does not require any pose annotations. Experimental results show our method can achieve 6.4%, 16.2%, and 3.9% performance improvement over the prior art OVE6D (w/o ICP) on LINEMOD, Occluded LINEMOD and T-LESS datasets respectively, and is comparable with other state of the art methods even the supervised ones. Compared with point-to-plane ICP, our method has the obvious advantage on computation speed, due to the merit of full play to the high parallel characteristics of deep learning based on GPU acceleration.

1 INTRODUCTION

Estimating the 6D pose of an object that includes 3D rotation and 3D translation, plays an important role in many real-world applications, such as robotic manipulation (Collet et al., 2011), 3D scene understanding (Huang et al., 2018), and augmented reality (Marchand et al., 2015). The object pose estimation task aims at predicting the geometric mapping from the camera coordinate system to the object coordinate system (Hodan et al., 2018), relying on the sensory input. This is a challenging and active problem, receiving extensive attentions from both academia and industry.

In 6D pose estimation, two main data modalities for robotic visual perception are color and depth, the former contains structure and texture information, while the latter contains surface geometry information. RGB-based 6D pose estimation attempts to establish sparse or dense 2D-3D correspondences between the 2D coordinates in the RGB image and the 3D coordinates on the object 3D model surface (Sundermeyer et al., 2018; Peng et al., 2019; Hodaň et al., 2020). Methods along this line achieve impressive performance when objects with rich textures, which however are not robust enough against typical challenges such as dynamic changes of the environment illumination. To overcome the limitation of RGB-only, many methods turn to take full advantage of multi-modal information. For instance, works (Sundermeyer et al., 2018; Hodaň et al., 2020) first perform the initial pose estimation based on RGB images and then further refine the result with depth images using geometry-based optimization.

Depth images contain rich geometric information of the object shape, which is helpful for inferring the object pose. Recently, with the widespread use of depth cameras and LiDAR, depth information becomes the main force for pose estimation instead of just serving as auxiliary information of RGB. Some methods emerge to use depth information represented by point clouds for deep learning based 6D object pose estimation. CloudPose proposed by Gao *et al.* (Gao et al., 2020) attempts to exploit 3D point clouds generated from object depth to perform 6D pose regression in a supervised manner. To alleviate the dependence on supervision information that is usually expensive to obtain, Gao *et al.* further propose CloudAAE (Gao et al., 2021) to perform network training on synthetic depth data. In both CloudPose and CloudAAE, the 6D pose estimated by deep learning is further refined with

Sensory Input	Method (+ICP refinement)	Occ. LINEMOD	T-LESS	Avg. Run.	ICP Avg. Run.	Ours Avg. Run.
Point Clouds	OVE6D (Cai et al., 2022)	55.3 (+15.9)	69.4 (+5.4)	50ms	50ms∼60ms	8ms ∼ 10ms
	CloudAAE (Gao et al., 2021)	57.1 (+6.1)	-	70ms		
RGB-D	AAE (Sundermeyer et al., 2018)	-	19.3 (+49.3)	23ms		
	PVNet (Peng et al., 2019)	42.4 (+26.6)	-	40ms		
	MP-Encoder (Sundermeyer et al., 2020a)	-	20.5 (+49.0)	200ms		
	PoseCNN (Xiang et al., 2017)	24.9 (+53.1)	-	-		

Table 1: We report the initial pose performance of the methods and the improvement with ICP refinement in brackets. Obviously, ICP can further improve the performance with a large margin and thus the refinement process is critical in real applications. We report ADD(S) on Occluded Linemod and VSD on T-LESS. The detail of metric can be found in Sec. 4.3.

a geometry-based optimization process to produce the final estimation result. Although achieving great progress, the existing point cloud based methods are not efficient in terms of run-time, as well as are restricted to the requirement for large amounts of pose annotations.

As stated above, the state-of-the-art RGB-D and point cloud based 6D pose estimation methods take the geometry-based optimization as the refinement stage. It has been demonstrated that using a pose refinement stage after the initial pose estimator is effective for 6D object pose estimation (Iwase et al., 2021a). The iterative closest point (ICP) is the most popular refinement strategy, which has been used in numerous works, such as (Xiang et al., 2017; Sundermeyer et al., 2018; Peng et al., 2019; Sundermeyer et al., 2020b; Hodaň et al., 2020; Gao et al., 2020; 2021; Cai et al., 2022), just to mention a few. We summarize their performance comparison with and without the ICP refinement procedure in Table 1, from which it can be found that the refinement stage plays a very important role in performance boosting. However, such performance improvement is achieved at the price of much more running time cost, since ICP suffers from slow convergence. Although some efficient variants of ICP have been proposed, such as point-to-plane ICP and plane-to-plane ICP (Rusinkiewicz & Levoy, 2001), the nature of iterative nonlinear optimization limit its efficiency.

In this paper, we present a simple yet efficient self-supervised point clouds alignment method via implicit neural network, which can serve as an alternative of ICP to achieve fast and accurate pose refinement. Our approach is inspired by the concept of signed distance function (SDF), which represents the surface of a shape by a continuous volumetric field. For a given spatial point, SDF outputs the distance of this point to the closest surface. In this way, the underlying surface is implicitly represented as the zero-level-set, *i.e.*, $SDF(\cdot) = 0$. Our key idea is to encode the surface of target point cloud into a SDF; the optimal rigid transformation then can be derived by addressing a minimization problem over the SDF. The SDF is specifically formed as an implicit neural network (INN). Implicit neural representation (Michalkiewicz et al., 2019) is a powerful tool to represent a 3D scene, which can be used to predict predefined local properties of any 3D spatial position. Moreover, it is a continuous representation, possessing stronger presentation ability compared with discrete representations such as mesh, volume or point cloud. We train INN in a self-supervised learning manner. Following the same setting as OVE6D (Cai et al., 2022), we assume that the 3D mesh model of the target object is accessible. Based on the target mesh model, we can construct a pretext task, where the supervision information is easy to obtain, to train the INN. Finally, using the derived SDF as the objective function, we obtain the optimal transformation with auto-grad tools (Paszke et al., 2019).

We conduct extensive experiments on three benchmark dataset: LineMOD (Wang et al., 2017), LineMOD-occlusion and TLESS (Hodan et al., 2017) dataset, where we choose OVE6D (Cai et al., 2022) that is most recently proposed SOTA method as the base stage and use our strategy for pose estimation refinement. The workflow of our method does not require any pose annotations. Experimental results show our method can achieve 6.4%, 16.0%, and 3.9% performance improvement over OVE6D on these datasets respectively, and is comparable with other SOTA methods even the supervised ones. Compared with point-to-plane ICP, our method has the obvious advantage on running time, which allows full play to the high parallel characteristics of deep learning based on GPU acceleration.

The main contributions of our method are summarized as follows:

- We propose a simple yet efficient strategy for 6D object pose refinement without any requirement on pose annotations, which achieves comparable accuracy with ICP while with much faster running speed.
- We propose to implicitly encode the geometry information of the target 3D model as a SDF. The task of 6D pose estimation is then reformulated as a minimization problem over the SDF, which can be efficiently addressed by back propagation.
- Extensive experiment results are provided to demonstrate that our strategy achieves significant improvement over the base method and achieves comparable results with current SOTA methods. Our refinement strategy gets at least 6 times faster than point-to-plane ICP.

2 Related work

2.1 6D pose estimation and refinement

Depth-based methods usually have satisfied performance. Traditional methods, such as Point Pair Feature-based methods (Drost et al., 2010) build feature descriptors for model and scene and generate corresponding to retrieve pose by comparing the similarity of the descriptors. Deep learning methods, such as CloudAAE (Gao et al., 2021) proposes a framework for directly regressing the 6D object pose, which can predict pose more effectively. RGB-based methods have lower performance than depth-based methods since it lacks spatial information. Most RGB-based methods focus on finding the correspondence between the pixel in 2D image and the point in 3D model and solve with Perspective-n-Point (PnP) algorithms (Lepetit et al., 2005; 2009). Methods with RGBD data can achieve more impressive performance by directly leverage two modalities data (He et al., 2021; 2020).

The initial predicted poses are usually not sufficient for direct use in applications. Thus refinement to further improve the performance is required. Most representative methods are Iterative Closet Points (ICP) and its variants (Zhang et al., 2021; Bouaziz et al., 2013). Learning-based methods are also proposed to solve the task. Several point cloud registration methods (Aoki et al., 2019; Choy et al., 2020) are proposed to solve the limits of the ICP, such as slow convergence and robustness to outliers. These methods usually need 3D mesh models and do the refinement with point cloud data. So methods without 3D mesh models or point cloud data have also been explored. Repose (Iwase et al., 2021b) proposes a deep texture render containing learn able parameter to align the input RGB image feature extracted by neural network and thus retrieve the pose with LM algorithm. CATRE (Liu et al., 2022) propose a refinement methods without requirement of 3D model in category level setting.

Although the learning-based methods have made great progress with higher computational efficiency, most of them require expensive 6D pose annotations. In view of this, methods without the annotations has been proposed recently. OVE6D (Cai et al., 2022) is a general object pose estimation method without 6D pose annotations from any dataset. It decomposes the 6D pose into viewpoint, in-plane rotation around the camera optical axis and translation, and introduces novel lightweight modules for estimating each component in a cascaded manner. For 6D pose estimation refinement, leaning-based methods without pose annotations are hardly mentioned. In this paper, we discuss the penitential of implicit neural network and adopt it to do 6D pose refinement task. The experiment results will show the effectiveness of the method.

2.2 Implicit neural representation

Image, video, and voxel as records of the world all are discrete mapping from the 2D/3D position to the pixel value. However, the real world should be continuous or approximately continuous. But it is difficult to save or formulate such mappings before the rise of deep learning. The neural network has proved its ability to approximate any function (Hornik et al., 1989). Thus, lots of attention has been paid to leveraging the network to represent the continuous mappings and it is named as implicit neural network.

Nerf (Mildenhall et al., 2020; Barron et al., 2021) is one of the most representative methods. The target task of it is novel view synthesis, which means predicting color images from any view given a few images. It uses the MLP network to predict color density and accumulation transparency of

3D spatial position. Then Nerf renders the color image following the traditional volume rendering procedure. Other works to leverage the implicit neural network to predict continuous SDF that implicitly indicates the surface of the object have also made great progress in 3D model reconstruction (Park et al., 2019; Takikawa et al., 2021).

The implicit neural network has proved its great potential in reconstruction work. But the development of such technology has made it not limit to the reconstruction. In this paper, we fully discuss and present the advantages of implicit neural work for the 6D pose estimation refinement task.

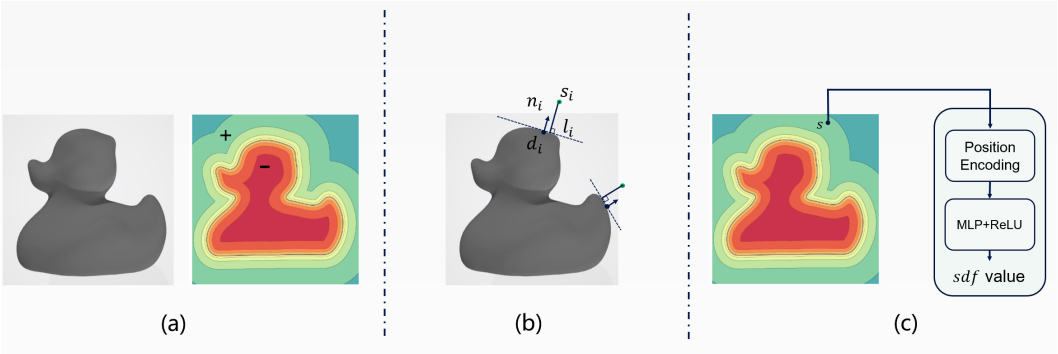


Figure 1: (a) is visualization of sdf value in space, which is presented by series of contour. (b) and (c) are the way of ICP and implicit neural network for calucating the distance from the query points to the nearest target surface.

2.3 POINT-TO-PLANE ICP

Point-to-Plane ICP is one of the most popular traditional pose refinement method with point cloud data. It can greatly improve the performance of the basement methods. It estimates the rigid transformation $\mathbf{R}, \mathbf{t}$ between the source point cloud $\{s_i | s_i \in \mathcal{S}\}$ and the target point cloud $\{d_i | d_i \in \mathcal{D}\}$ with following optimization objective:

$$\arg\min_{\mathbf{R},\mathbf{t}} \sum_i ||(\mathbf{R} * s_i + \mathbf{t} - d_i) \cdot n_i||_2. \tag{1}$$

where n_i is the surface normal of the target point cloud at d_i. It constrains the distance between the point of the source point cloud and the surface of the target point cloud. It calculates the projection on the surface normal. It could be solved by formulating the objective to a least square problem and thus obtaining a closed-form solution. The main drawbacks are calculating the surface normals is time-consuming, the close-form solution needs SVD decomposition, which is unstable and sensitive to outliers and the linear approximation of the rotation matrix is inappropriate with the large-errored pose. Our proposed method following similar procedure, which optimizes the distance between the points of source point cloud and the surface of the target point cloud. We leverage a neural network to fast and accurate compute the distance. A clearly comparison is illustrated in Fig. 1. Our method solve the limits of ICP, and thus has great potential to the 6D pose estimation refinement task.

3 METHOD

In point cloud based 6D pose estimation, given a known object represented by a point cloud in the camera coordinate C, the aim is to find the rotation $\mathbf{R}$ and the translation $\mathbf{t}$ that describe the rigid transformation between the object coordinate system O and C. Specifically, defining the source point cloud in C as $\{s_i | s_i \in \mathcal{S}\}$ and the target point cloud in O as $\{d_i | d_i \in \mathcal{D}\}$, estimation of $\mathbf{R} \in so(3)$ and $\mathbf{t} \in \mathbb{R}^3$ from the source to the target can be formulated as:

$$\mathbf{R}^*, \mathbf{t}^* = \arg\min_{\mathbf{R},\mathbf{t}} \sum_i \mathcal{G}\left(\mathbf{R} * s_i + \mathbf{t}\ , f(s_i, \mathcal{D})\right). \tag{2}$$

where $f(\cdot)$ denotes the operation of finding the nearest surface of s_i in the target point cloud; $\mathcal{G}(\cdot)$ presents Euclidean distance between the point from source point cloud and the point or surface of target point cloud.

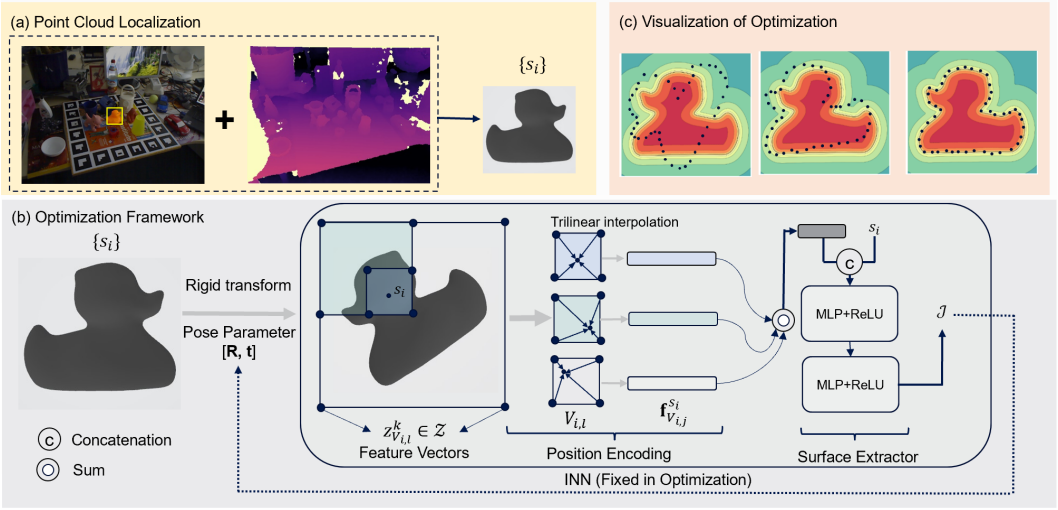


Figure 2: Overall of proposed scheme. (a) Point Cloud Registration. We localize the point cloud with segmentation mask predicted by Mask-RCNN (He et al., 2017) (b) Proposed 6D pose refinement framework. The implicit neural network is off-line trained and remains fixed during the refinement procedure. We iteratively optimize the SDF predicted by the network to retrieve the pose. (c) Visualization of Optimization. The transformed points will get close to the target object surface with the optimization process.

The proposed scheme is illustrated in Fig. 2. We first localize the source point cloud $\mathcal{S}$ with the segment mask predicted by Mask-RCNN (He et al., 2017). For the target 3D model, an implicit neural networks $f_\theta(\cdot)$ is trained offline to encode accurate 3D geometry by learning the signed distance function (SDF), which defines the surface by its zero level-set. In online pose estimation, $f_\theta(\cdot)$ is fixed and used to approximate the calculation process of $g(\cdot)$. The minimization problem formulated in Eq. 2 is iteratively optimized with auto-grad tool to derive the optimal $\mathbf{R}^*$ and $\mathbf{t}^*$. In the following, we will elaborate the main modules.

3.1 Neural Signed Distance Function

A signed distance function (SDF) $h : \mathbb{R}^3 \to \mathbb{R}$ is a continuous mapping function. For a given spatial point $\mathbf{x} \in \mathbb{R}^3$, $h(\mathbf{x})$ outputs the closest distance to the object surface, whose sign indicates whether $\mathbf{x}$ is inside (negative) or outside (positive) of the object (Park et al., 2019). The underlying surface $\mathbf{S}$ of an object thus can be implicitly represented as the zero level-set of h:

$$\mathbf{S} = \{\mathbf{x} \in \mathbb{R}^3 | h(\mathbf{x}) = 0\}. \tag{3}$$

Considering that the point clouds are points on the surface of objects, their corresponding SDF values should be zero. Accordingly, we encode the surface geometry of the target 3D model by SDF. We leverage an implicit neural network f_θ to approximate the SDF h. Following the same setting as OVE6D (Cai et al., 2022), we assume that the 3D mesh model of the target object is accessible, according to which we can construct a large amounts of $\{\mathbf{x}, d\}$ pairs, where $\mathbf{x}$ is sampled points in O and d is calculated by $h(\mathbf{x}) = d$. The next step is to directly let the network learn to regress the continuous SDF from point samples. In the following, we introduce in detail the network architecture and training procedure.

Network Architecture: The network mainly consist of position encoding module $PE(\cdot)$, surface extracting modules $SE(\cdot)$ and a collection of feature vectors $\mathcal{Z}$, following (Takikawa et al., 2021). $\mathcal{Z}$ is organised with a sparse voxel octree (SVO) spanning the the space $\mathcal{B}$ in object coordinate as illustrated in Fig. 2. Each voxel $V_{i,l}$ where the subscripts indicate i-th voxel in l-th resolution level in the octree holds 8 learnable feature parameter $\{z^k_{V_{i,l}}\}_{k=1...8} \subset \mathcal{Z}$ at its corner. Given a query point $\mathbf{x}$ and target resolution level L, $PE(\cdot)$ will search for voxels containing $\mathbf{x}$ with resolution no more than L. Then it computes the features $\mathbf{f}^{\mathbf{x}}_{V_{i,l}}$ of $\mathbf{x}$ for each voxel by trilinearly interpolating the

Algorithm 1: Proposed Framework

Input: Implicit neural network $f_\theta(\cdot)$ with max resolution level L_{max}; Initial pose $\mathbf{R}_o, \mathbf{t}_o$ Rotation learning rate λ_1; Transition learning rate λ_2; Iteration number n; 3D mesh model $\mathcal{O}$; Source point cloud $\{s_i\}$

Output: Refined pose parameter $\mathbf{R}^*, \mathbf{t}^*$

Formulate pairs training pairs of 3D position space and SDF value $(\mathbf{x}, d)$
Train the implicit network f_θ.
  $\theta^* = \arg\min_\theta \mathbb{E}_{\mathbf{x},d} \sum_{L=1}^{L_{max}} \mathcal{L}\left(d_L, d\right)$
Initialize pose parameter: $\mathbf{R} = \mathbf{R}_o$, $\mathbf{t} = \mathbf{t}_o$
for $i = 1...n$ **do**
  Calculate objective: $\mathcal{J} = \sum_i |f_\theta^*(\mathbf{R} * s_i + t)|$
  Calculate gradients: $\Delta\mathbf{R} = \frac{\partial \mathcal{J}}{\partial \mathbf{R}}, \quad \Delta\mathbf{t} = \frac{\partial \mathcal{J}}{\partial \mathbf{t}}$
  Update with Adam optimizer: $\mathbf{R} \leftarrow \mathbf{R} + \lambda_1 \mathcal{A}(\Delta\mathbf{R}), \quad \mathbf{t} \leftarrow \mathbf{t} + \lambda_2 \mathcal{A}(\Delta\mathbf{t})$
end
Output pose parameter $\mathbf{R}^* = \mathbf{R}$, $\mathbf{t}^* = \mathbf{t}$

corner features:

$$\mathbf{f}^{\mathbf{x}}_{V_{i,l}} = \rho(\mathbf{x}, \{z^k_{V_{i,l}}\}). \tag{4}$$

where $\rho(\cdot)$ denotes interpolate operation. $PE(\cdot)$ will sum the features across the voxels and concat them with query points $\mathbf{x}$ as outputs:

$$PE(\mathbf{x}, L) = [\mathbf{x}, \sum_{l=1}^{L} \mathbf{f}^{\mathbf{x}}_{V_{i,l}}]. \tag{5}$$

Passing the results of $PE(\cdot)$ to the surface extractor $SE(\cdot)$ which consists of a combination of several MLP layer and ReLU layer, it will predict the final SDF value with the query resolution level L:

$$d_L = SE(PE(\mathbf{x}, L)). \tag{6}$$

During the inference stage, the whole network will directly output the SDF value with max resolution level L_{max}: $f_\theta(\mathbf{x}) = d_{L_{max}}$.

Training Procedure: Given the sampled pair $(\mathbf{x}, d)$, the loss function of the implicit neural network is:

$$\theta^* = \arg\min_\theta \mathbb{E}_{\mathbf{x},d} \sum_{L=1}^{L_{max}} \mathcal{L}\left(d_L, d\right). \tag{7}$$

It calculates the prediction error across multi-level resolution. The parameters θ of INN contain the feature vectors $\mathcal{Z}$ and the MLP layer in surface extractor module. They are optimized with Adam optimizer (Kingma & Ba, 2014) and the whole process only needs about 10 minutes.

3.2 6D POSE ESTIMATION REFINEMENT FRAMEWORK

According to Sec. 3, we have obtained a well trained implicit neural network f_{θ^*}. We leverage it to calculate the distance between the points of source point cloud and surface of target object point cloud. We define the target optimized objective as:

$$\mathbf{R}^*, \mathbf{t}^* = \arg\min_{\mathbf{R},\mathbf{t}} \mathcal{J}(\mathbf{R}, \mathbf{t}) = \arg\min_{\mathbf{R},\mathbf{t}} \sum_i |f_{\theta^*}(\mathbf{R} * s_i + t)|. \tag{8}$$

We solve the objective iteratively with initial pose provided by OVE6D (Cai et al., 2022) . We calculate the gradients of the objective with auto-grad tool (Paszke et al., 2019). The network's parameters remain fixed during the whole procedure. Finally, we update rotation and transition separately with Adam optimizers (Kingma & Ba, 2014) using the gradients and different learning rate λ_1, λ_2:

$$\mathbf{R} \leftarrow \mathbf{R} + \lambda_1 \mathcal{A}(\frac{\partial \mathcal{J}}{\partial \mathbf{R}}), \quad \mathbf{t} \leftarrow \mathbf{t} + \lambda_2 \mathcal{A}(\frac{\partial \mathcal{J}}{\partial \mathbf{t}}). \tag{9}$$

4 EXPERIMENT

4.1 IMPLEMENTATION DETAILS

All the experiments are implemented using PyTorch (Paszke et al., 2019). We trained the implicit neural network with the pairs $(\mathbf{x}, d)$ as described in the method. The training time is around 10 minutes on a single Titan V gpu. For OVE6D, we use the pretrained model trained on ShapeNet (Chang et al., 2015) which contains 12490 objects. And following OVE6D, we leverage Mask-RCNN (He et al., 2017) to localize the target's point cloud. In the refinement procedure, we adopt Adam (Kingma & Ba, 2014) optimizers to auto-optimize the rotation and transition separately.

4.2 DATASET

We evaluate our method on LINEMOD (Wang et al., 2017), Occluded LINEMOD, and T-Less dataset (Hodan et al., 2018)). LINEMOD has 13 objects, 12 scenes, and about 18K images. We follow the test protocol from the previous work (Wang et al., 2019). Occluded LNEMOD is a subset of LINEMOD but is more challenging containing 8 objects with heavy occlusions. We evaluate our performance of Occluded LINEMOD on its 1214 testing images. And T-Less dataset includes 30 texture-less and symmetric industrial objects with highly similar shapes. We evaluate the performance on the PrimeSense test set following the procotol specified in the BOP challenge (Hodan et al., 2018). Following OVE6D (Cai et al., 2022), we report ADD(S) score on LINEMOD and Occluded LINEMOD and VSD score on T-Less.

4.3 EVALUATION METRICS

For Linemod and Linemod occlusion, the average distance metrics ADD and ADDS are used usually. For object $\mathcal{O}$ containing N vertex $\{v_i\}_{i=1...N}$, given the predicted pose $\mathbf{R}, \mathbf{t}$ and ground truth pose $\mathbf{R}^*, \mathbf{t}^*$ ADD is calculated as:

$$ADD = \frac{1}{N} \sum_{v_i \in \mathcal{O}} ||(\mathbf{R} * v_i + \mathbf{t}) - (\mathbf{R}^* * v_i + \mathbf{t}^*). \tag{10}$$

For symmetric objects, ADDS is calcuated as :

$$ADDS = \frac{1}{N} \sum_{v_i \in \mathcal{O}} \min_{v_j \in \mathcal{O}} ||(\mathbf{R} * v_1 + \mathbf{t}) - (\mathbf{R}^* * v_2 + \mathbf{t}^*)||. \tag{11}$$

For TLess dataset, visible surface discrepancy(VSD) is userd for evaluating the pose. Use the predicted pose and ground truth pose to render depth maps $\hat{S}$ and $\bar{S}$ and masks $\hat{V}$ and $\bar{V}$. The pose error e_{VSD} is caculated as:

$$e_{\text{VSD}}\left(\hat{S}, \bar{S}, S_I, \hat{V}, \bar{V}, \tau\right) = \underset{p \in \hat{V} \cup \bar{V}}{\text{avg}} \begin{cases} 0 & \text{if } p \in \hat{V} \cap \bar{V} \wedge |\hat{S}(p) - \bar{S}(p)| < \tau \\ 1 & \text{otherwise} \end{cases}. \tag{12}$$

And given a predefined error threshold θ, the final VSD score is:

$$VSD = \frac{\sum e_{VSD} < \theta}{N}. \tag{13}$$

4.4 COMPARISON WITH SOTA AND ICP

We compare our method with a branch of SOTA methods on LINEMOD, Occluded LINEMOD and T-LESS in Table 2. And we list the performance on each object in the dataset in Table 3 and Table 4 and provide qualitive results in Fig. 3 for a fully comparison between our proposed framework and ICP.

Occluded Linemod: This dataset is a challenging dataset with heavy occlusion. But our method has an improvement of 16.2% over OVE6D and outperforms FFB6D which is a supervised method with a large margin as shown in Table 2. The results in Table 4 and Fig. 3 further prove the generality of our framework under the situation of heavy occlusion are comparable with ICP.

Dataset	Method	Sensor data type	User anno.	with ref.	Results
Occluded LINEMOD	FFB6D (He et al., 2021)	RGBD	True		66.2
	RePose (Iwase et al., 2021b)	RGB	True		51.6
	CloudAAE(GT) (Gao et al., 2021)	D		True	63.2
	OVE6D(MRCNN) (Cai et al., 2022)	D			55.3
	OVE6D(MRCNN) (Cai et al., 2022)	D		True	<u>71.2</u>
	OVE6D+Ours	D			**71.5**
LINEMOD	FFB6D (He et al., 2021)	RGBD	True		**99.7**
	RePose (Iwase et al., 2021b)	RGB	True		<u>96.1</u>
	CloudAAE(GT) (Gao et al., 2021)	D		True	92.5
	OVE6D(MRCNN) (Cai et al., 2022)	D			86.1
	OVE6D(MRCNN) (Cai et al., 2022)	D		True	92.4
	OVE6D+Ours	D			92.5
T-Less	StablePose (Shi et al., 2021)	D	True		73.0
	CopyPose (Labbé et al., 2020)	RGB	True		63.8
	MP-Encoder (Sundermeyer et al., 2020a)	RGBD	True		69.5
	OVE6D(MRCNN) (Cai et al., 2022)	D			69.4
	OVE6D(MRCNN) (Cai et al., 2022)	D		True	**74.8**
	OVE6D+Ours	D			<u>73.3</u>

Table 2: Evaluation on LINEMOD, Occluded LINEMOD and T-Less. We report the average ADD(-S) recall on LINEMOD and Occluded LINEMOD and the average VSD recall on T-Less. We highlight the best and the second best methods in bold and underline in each group. MRCNN and GT indicate using the segmentations masks provided by Mask-RCNN and the ground truth.

LINEMOD: As shown in Table 2 and Table 3, Our framework has an improvement of 6.4% over OVE6D and is comparable with the annotation-free SOTA method CloudAAE (Gao et al., 2021) while their method needs ground truth segmentation masks and ours uses masks predicted by Mask-RCNN (He et al., 2017), which is more close to real applications. FFB6D (He et al., 2021) achieves best performance with fully exploring RGB and depth information. It should be noticed that FFB6D needs real pose annotations, which is expensive and challenging to collect. Overall, our refinement framework can improve OVE6D to the satisfied results for real application. ICP can also improve the performance of OVE6D to SOTA but with higher computation cost. As shown in Table 3 and Fig. 3, Our framework guarantees both the performance on each object and high inference speed as shown in Table 1.

T-Less: This dataset has large amounts of texture-less and symmetric objects, which is also challenging and difficult for pose estimation. OVE6D with ICP refinement outperforms the supervised method StablePose and CopyPose as presented in Table 2. And we have comparable results with it, which further illustrates the ability of generalizing to the complicate scenes and objects of our proposed framework.

Object name	ape	bench.	cam	can	cat	driller	duck
OVE6D w/o ICP	91.3	98.5	60.4	92.5	90.1	77.7	80.4
OVE6D w/ ICP	95.4	99.7	73.9	96.5	97.9	94.1	95.0
OVE6D+Ours	96.1	100.0	74.1	97.0	97.7	93.4	95.3
Object name	**egg.**	**glue**	holep.	iron	lamp	phone	Avg.
OVE6D w/o ICP	93.6	93.6	83.4	95.3	79.1	82.8	86.1
OVE6D w/ ICP	93.6	94.3	86.0	97.7	87.2	89.8	92.4
OVE6D+Ours	93.6	93.7	86.4	97.7	87.6	90.3	92.5

Table 3: Compared with ICP on LINEMOD. We report ADD and ADDS score for each object. The bold text indicates symmetric object.

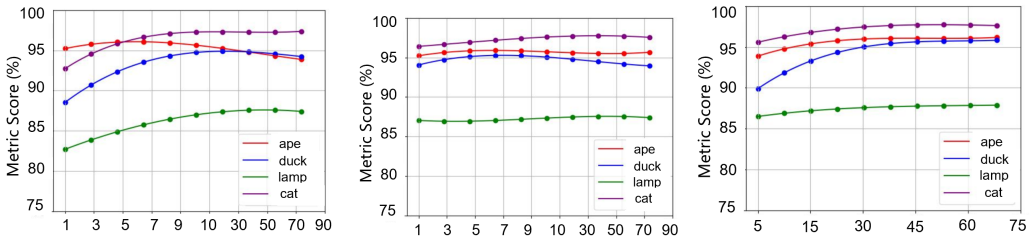
Object name	ape	can	cat	driller	duck	eggbox	glue	holep.	Avg.
OVE6D w/o ICP	48.5	59.8	39.4	27.9	51.9	66.7	68.9	79.6	55.3
OVE6D w/ ICP	66.9	80.9	55.4	69	67.8	68.1	71.1	90.6	71,2
OVE6D+Ours	67.5	82.7	53.6	69.1	69.6	68.4	71.5	89.6	71.5

Table 4: Compared with ICP on Occluded LINEMOD. We report ADD and ADDS score for each object. The bold text indicates symmetric object.

(a) OVE6D (b) ICP (c) SDFR

(d) OVE6D (e) ICP (f) SDFR

Figure 3: Qualitative results on Linemod and Linemod occlusion. The first raw and the second raw present objects in Linemod and Linemod occlusion, respectively.



(a) LR of rotation (1e-3) (b) LR of transition (1e-4) (c) Iteration number

Figure 4: Ablation study on the hyper-parameter of our framework.

4.5 Ablation Study

We build our ablation study on the learning rate of rotation, transition and iteration number, which is illustrated in Fig.4. Fig.4(a) and Fig.4(b) indicate the learning rate of rotation and the learning rate of transition should be different. Each figure indicates our framework's performance remains stable with a wide range of proper learning rate. In view of this, the hyper-parameters are easy to tune and thus the refinement framework is more convenience to use in real applications.

5 Conclusion

In this paper, we presented a fast 6D pose estimation refinement strategy, which works as an alternative of the popular iterative closest point (ICP) to achieve fast and accurate pose refinement. Our work is inspired by the observation that pose refinement after the initial pose estimator plays a very important role in performance boosting while the ICP-based refinement suffers from slow convergence. We propose to encode the surface of target point cloud into a signed distance function, which further serves as the minimization objective function to derive the optimal rigid transformation. Extensive experimental results are provided to demonstrate the superiority of our method.

REFERENCES

Yasuhiro Aoki, Hunter Goforth, Rangaprasad Arun Srivatsan, and Simon Lucey. Pointnetlk: Robust & efficient point cloud registration using pointnet. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 7163–7172, 2019.

Jonathan T Barron, Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, and Pratul P Srinivasan. Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5855–5864, 2021.

Sofien Bouaziz, Andrea Tagliasacchi, and Mark Pauly. Sparse iterative closest point. In *Computer graphics forum*, volume 32, pp. 113–123. Wiley Online Library, 2013.

Dingding Cai, Janne Heikkilä, and Esa Rahtu. Ove6d: Object viewpoint encoding for depth-based 6d object pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6803–6813, 2022.

Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015.

Christopher Choy, Wei Dong, and Vladlen Koltun. Deep global registration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2514–2523, 2020.

Alvaro Collet, Manuel Martinez, and Siddhartha S Srinivasa. The moped framework: Object recognition and pose estimation for manipulation. *The international journal of robotics research*, 30 (10):1284–1306, 2011.

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*, pp. 998–1005. Ieee, 2010.

Ge Gao, Mikko Lauri, Yulong Wang, Xiaolin Hu, Jianwei Zhang, and Simone Frintrop. 6d object pose regression via supervised learning on point clouds. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3643–3649. IEEE, 2020.

Ge Gao, Mikko Lauri, Xiaolin Hu, Jianwei Zhang, and Simone Frintrop. Cloudaae: Learning 6d object pose regression with on-line data synthesis on point clouds. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 11081–11087. IEEE, 2021.

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

Yisheng He, Wei Sun, Haibin Huang, Jianran Liu, Haoqiang Fan, and Jian 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*, pp. 11632–11641, 2020.

Yisheng He, Haibin Huang, Haoqiang Fan, Qifeng Chen, and Jian 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*, pp. 3003–3013, 2021.

Tomáš Hodan, Pavel Haluza, Štepán Obdržálek, Jiri Matas, Manolis Lourakis, and Xenophon Zabulis. T-less: An rgb-d dataset for 6d pose estimation of texture-less objects. In *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 880–888. IEEE, 2017.

Tomas Hodan, Frank Michel, Eric Brachmann, Wadim Kehl, Anders GlentBuch, Dirk Kraft, Bertram Drost, Joel Vidal, Stephan Ihrke, Xenophon Zabulis, et al. Bop: Benchmark for 6d object pose estimation. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 19–34, 2018.

Tomáš Hodaň, Dániel Baráth, and Jiří Matas. EPOS: Estimating 6D pose of objects with symmetries. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5):359–366, 1989.

Siyuan Huang, Siyuan Qi, Yinxue Xiao, Yixin Zhu, Ying Nian Wu, and Song-Chun Zhu. Cooperative holistic scene understanding: Unifying 3d object, layout, and camera pose estimation. In *Advances in Neural Information Processing Systems*, pp. 206–217, 2018.

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 (ICCV)*, pp. 3303–3312, October 2021a.

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*, pp. 3303–3312, 2021b.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Yann Labbé, Justin Carpentier, Mathieu Aubry, and Josef Sivic. Cosypose: Consistent multi-view multi-object 6d pose estimation. In *European Conference on Computer Vision*, pp. 574–591. Springer, 2020.

Vincent Lepetit, Pascal Fua, et al. Monocular model-based 3d tracking of rigid objects: A survey. *Foundations and Trends® in Computer Graphics and Vision*, 1(1):1–89, 2005.

Vincent Lepetit, Francesc Moreno-Noguer, and Pascal Fua. Epnp: An accurate o (n) solution to the pnp problem. *International journal of computer vision*, 81(2):155–166, 2009.

Xingyu Liu, Gu Wang, Yi Li, and Xiangyang Ji. Catre: Iterative point clouds alignment for category-level object pose refinement. *arXiv preprint arXiv:2207.08082*, 2022.

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.

Mateusz Michalkiewicz, Jhony K Pontes, Dominic Jack, Mahsa Baktashmotlagh, and Anders Eriksson. Implicit surface representations as layers in neural networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4743–4752, 2019.

Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *European conference on computer vision*, pp. 405–421. Springer, 2020.

Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 165–174, 2019.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

Sida Peng, Yuan Liu, Qixing Huang, Xiaowei Zhou, and Hujun Bao. Pvnet: Pixel-wise voting network for 6dof pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4561–4570, 2019.

Szymon Rusinkiewicz and Marc Levoy. Efficient variants of the icp algorithm. In *Proceedings third international conference on 3-D digital imaging and modeling*, pp. 145–152. IEEE, 2001.

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

Martin Sundermeyer, Zoltan-Csaba Marton, Maximilian Durner, Manuel Brucker, and Rudolph Triebel. Implicit 3d orientation learning for 6d object detection from rgb images. In *The European Conference on Computer Vision (ECCV)*, September 2018.

Martin Sundermeyer, Maximilian Durner, En Yen Puang, Zoltan-Csaba Marton, Narunas Vaskevicius, Kai O Arras, and Rudolph Triebel. Multi-path learning for object pose estimation across domains. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 13916–13925, 2020a.

Martin Sundermeyer, Zoltan-Csaba Marton, Maximilian Durner, and Rudolph Triebel. Augmented autoencoders: Implicit 3d orientation learning for 6d object detection. *International Journal of Computer Vision*, 128(3):714–729, 2020b.

Towaki Takikawa, Joey Litalien, Kangxue Yin, Karsten Kreis, Charles Loop, Derek Nowrouzezahrai, Alec Jacobson, Morgan McGuire, and Sanja Fidler. Neural geometric level of detail: Real-time rendering with implicit 3d shapes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11358–11367, 2021.

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*, pp. 3343–3352, 2019.

Yue Wang, Shusheng Zhang, Sen Yang, Weiping He, Xiaoliang Bai, and Yifan Zeng. A line-mod-based markerless tracking approachfor ar applications. *The International Journal of Advanced Manufacturing Technology*, 89(5):1699–1707, 2017.

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.

Juyong Zhang, Yuxin Yao, and Bailin Deng. Fast and robust iterative closest point. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.