CATRE: Iterative Point Clouds Alignment for Category-level Object Pose Refinement

Xingyu Liu[1,*] 🔘   Gu Wang[2,*] 🔘   Yi Li[3] 🔘   Xiangyang Ji[1] 🔘

[1]Tsinghua University, BNRist   [2]JD.com   [3]University of Washington
liuxy21@mails.tsinghua.edu.cn, guwang12@gmail.com,
yili.matrix@gmail.com, xyji@tsinghua.edu.cn

Abstract. While category-level 9DoF object pose estimation has emerged recently, previous correspondence-based or direct regression methods are both limited in accuracy due to the huge intra-category variances in object shape and color, *etc*. Orthogonal to them, this work presents a category-level object pose and size refiner CATRE, which is able to iteratively enhance pose estimate from point clouds to produce accurate results. Given an initial pose estimate, CATRE predicts a relative transformation between the initial pose and ground truth by means of aligning the partially observed point cloud and an abstract shape prior. In specific, we propose a novel disentangled architecture being aware of the inherent distinctions between rotation and translation/size estimation. Extensive experiments show that our approach remarkably outperforms state-of-the-art methods on REAL275, CAMERA25, and LM benchmarks up to a speed of $\approx$85.32 Hz, and achieves competitive results on category-level tracking. We further demonstrate that CATRE can perform pose refinement on unseen category. Code and trained models are available.[1]

1   Introduction

Estimating the 6DoF pose, *i.e.*, 3DoF orientation and 3DoF localization, of an object in Euclidean space plays a vital role in robotic manipulation [49,13], 3D scene understanding [36,22] and augmented/virtual reality [35,46]. The vast majority of previous works [30,64,45,62,53,54,31,52,38] study with instance-level object pose estimation, which can be decomposed by two procedures: initial pose estimation and pose refinement. Despite considerable progress has been made in instance-level settings, the generalizability and scalability w.r.t. unseen objects are limited due to the high dependency of known exact CAD models during both training and inference.

To alleviate this problem, recently, increasing attention has been received on category-level 9DoF pose (*i.e.*, 6DoF pose and 3DoF size) estimation which aims to handle novel instances among known categories without requiring CAD models for test. As an early proposed work, Wang *et al.* [55] predict the normalized object coordinates (NOCS) in image space and then solve the pose and

---

[1]https://github.com/THU-DA-6D-Pose-Group/CATRE.git

size by matching NOCS against observed depth with Umeyama algorithm [50]. Afterwards, several works [47,7,56,14] attempt to deform the shape prior (*i.e.*, the mean shape) of a category towards observed instances to improve the accuracy of correspondences matching. However, those methods highly rely on the RANSAC procedure to remove outliers thus making them non-differentiable and time-consuming. Contrary to correspondence-based methods, some more recent works [9,32] propose to directly regress pose and size to achieve a higher speed during inference. Nevertheless, these end-to-end approaches are oftentimes error-prone in that they are not sufficiently aware of the inherent distinctions between rotation and translation/size estimation. To summarize, compared with the milestones achieved by state of the art in instance-level works, category-level pose estimation methods are still limited in accuracy.

Previously, when the CAD model is available, one common way of enhancing pose accuracy is to apply a post-refinement step through matching the rendered results against observed images given initial estimates, which has been widely explored in both traditional [2,44] and learning-based [30,59,27,25] methods. Being motivated by this, we seek to tackle the above problem by investigating object pose refinement at the category level. However, traditional object pose refinement methods rely on the CAD model to perform render-and-compare, which is not accessible when we conduct category-level object pose estimation.

To solve this dilemma, we propose a novel method for *CAT*egory-level object pose *RE*finement (*CATRE*), leveraging the abstract shape prior information instead of exact CAD models. As shown in Fig. 1, we aim to directly regress the relative pose and size transformations by aligning the partially observed point cloud against the transformed shape prior given initial pose and size predictions. In specific, we first use the initial pose prediction to focalize the shape prior and observed point cloud into a limited range. Then, a Siamese PointNet-based [40] encoder is employed to map the two input point clouds into a common feature space while maintaining relevant features for rigid transformation. Finally, we design two distinct heads to predict relative rotation and translation/size transformations in a disentangled manner. This is based on the observation that rotation is heavily reliant on point-level local geometry whereas translation and size reside with the object-level global feature. Besides, the procedure of refinement is conducted iteratively to achieve more accurate results.

Extensive experiments demonstrate that our proposed method can accurately yet efficiently refine and track category-level object poses. Exemplarily, we achieve a significant improvement over the baseline SPD [47] with an improvement of 26.8% on the strict $5°\,2\,\text{cm}$ metric and 14.5% on the IoU_{75} metric on REAL275 dataset.

To sum up, our contributions are threefold: i) To the best of our knowledge, we propose the first versatile pipeline for category-level pose refinement based on point clouds leveraging abstract shape prior, without requiring exact CAD models during training or inference. ii) For the learning of relative transformation, a pose-guided focalization strategy is proposed to calibrate input point clouds. We further introduce a novel disentangled architecture being aware of the inherent
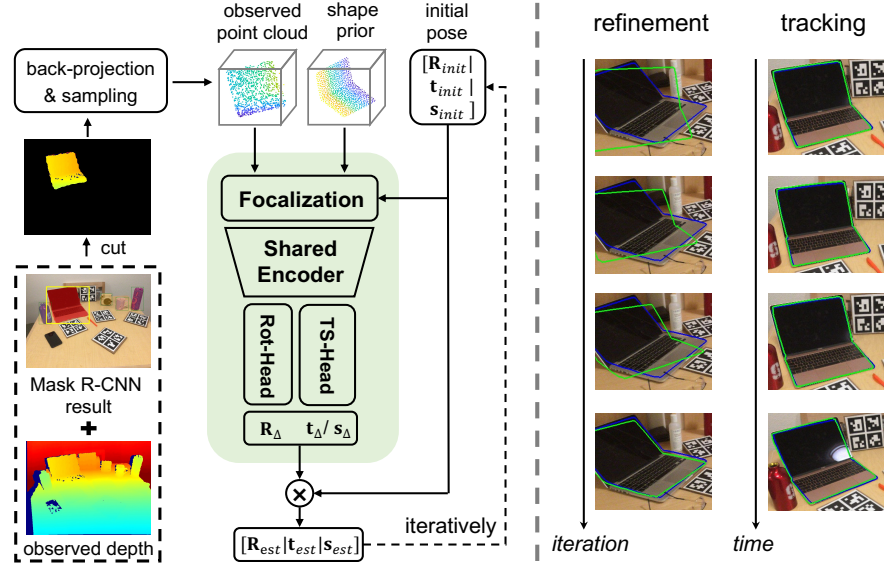
Fig. 1: The framework of CATRE. Given an initial pose and size estimate $[\mathbf{R}_{init}|\mathbf{t}_{init}|\mathbf{s}_{init}]$, CATRE predicts a relative transformation $[\mathbf{R}_{\Delta}|\mathbf{t}_{\Delta}|\mathbf{s}_{\Delta}]$ by iteratively aligning two focalized point clouds, *i.e.*, shape prior and observed point cloud. The framework can perform pose refinement and tracking, where blue and green contours reflect ground-truth and predicted poses, respectively.

attributions in pose estimation. Thanks to these key ingredients, we overcome the drawback of being error-prone while maintaining the high speed of direct regression. iii) Our versatile framework can also perform category-level pose tracking and achieve competitive performance against state of the art, whilst at 7x faster speed. Meanwhile, CATRE can be naturally extended to instance-level and unseen category pose refinement leveraging appropriate shape priors.

2   Related Work

Category-level Object Pose Estimation Category-level pose estimation aims to predict the 9DoF pose of a novel instance without the aid of its CAD model. Existing works can be generally categorized to correspondence-based [55,7,47,56] and direct regression [32,9,6] approaches. Correspondence-based approaches first predict dense object coordinates in a normalized canonical space (NOCS) [55] and then solve the pose by Umeyama algorithm [50]. Notably, SPD [47] proposes to extract a categorical shape prior and adapt it to various instances via deformation prediction, in an effort to improve the matching of correspondences. On the other hand, direct regression methods predict object pose in an end-to-end manner, achieving a higher inference speed. For instance, FSNet [9] decouples

rotation prediction into two orthogonal axes estimation. DualPoseNet [32] exploits spherical convolutions to explicitly regress the pose with an auxiliary task of canonicalizing the observed point cloud meanwhile.

While this field has emerged recently, the accuracy of estimating category-level pose is still far from satisfactory compared with instance-level settings. Orthogonally, this work builds an end-to-end category-level pose refinement pipeline which can largely enhance the performance whilst being fast.

Object Pose Refinement In instance-level cases [30,27,59,64,45,62,53], object pose refinement has demonstrated to be very effective. Notably, DeepIM [30] predicts pose transformation by comparing the rendered image against the observed image. se(3)-TrackNet [59] further leverages this render-and-compare strategy to object tracking using RGB-D images. Extended from DeepIM, CosyPose [27] conquers the leaderboard of BOP Challenge 2020 [21], showing the powerful capability of the render-and-compare technique.

Different from the instance-level settings, category-level pose refinement is still rarely explored. DualPoseNet [32] refines their prediction by a self-adaptive pose consistency loss, which is only applicable to specific networks. However, our work is generalizable to various kinds of baselines [55,47,32] without re-training.

Category-level Object Pose Tracking 6-PACK [51] performs category-level tracking by anchor-based keypoints generation and matching between adjacent frames. CAPTRA [60] tracks rigid and articulated objects by employing two separate networks to predict the interframe rotation and normalized coordinates respectively. Recently, iCaps [12] leverages a particle filtering framework to estimate and track category-level pose and size. Note that CAPTRA could be adapted for refinement in theory, although there are no relevant experiments. Still, there might be some limitations. First, it relies on CAD models to provide supervision signals (*i.e.*, NOCS map) for training, while CATRE does not need exact CAD models during training or inference. Moreover, it was proposed for tracking where the errors between adjacent frames are minor, so it is unclear if CAPTRA could handle noises from other estimation methods. However, as CATRE directly regresses the relative pose, it can perform iterative inference with a faster speed and more robust performance w.r.t. noises.

Point Cloud Registration A closely related field to depth-based refinement is point cloud registration. Traditional methods like ICP [2] and its variants [3,44,42] require reliable pose initialization. To overcome this problem, learning-based methods are proposed recently [58,1,57,10]. Works like [16,48] use the non-overlapped prior as CATRE does, but they usually learn the deformation rather than the relative pose. Besides, shape prior in CATRE can be very different from the objects, varying from instance-specific keypoints to categorical mean shape to generic bounding box corners.
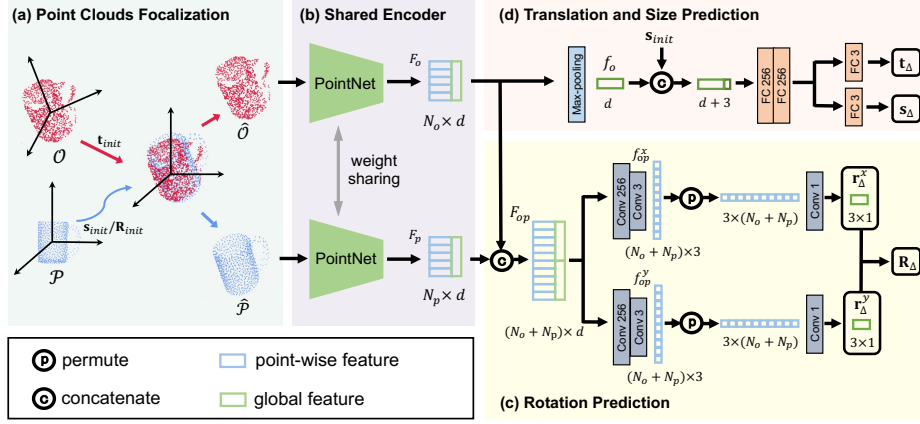
Fig. 2: The network architecture of CATRE. Taking an observed point cloud $\mathcal{O}$ and shape prior $\mathcal{P}$ as input, CATRE first focalizes them into a limited range using initial pose prediction $[\mathbf{R}_{init}|\mathbf{t}_{init}|\mathbf{s}_{init}]$. Then, we employ a Siamese geometric encoder followed by two disentangled heads to predict relative pose transformation $[\mathbf{R}_\Delta|\mathbf{t}_\Delta|\mathbf{s}_\Delta]$.

3 Category-level Object Pose Refinement

This section first addresses the problem formulation and framework overview (Fig. 1), and then describes the key ingredients of our approach (Fig. 2) for category-level object pose refinement. Afterwards, we present the training and testing protocol and show that CATRE can be naturally applied to pose tracking.

3.1 Problem Formulation

Given an initial pose and size estimate $[\mathbf{R}_{init}|\mathbf{t}_{init}|\mathbf{s}_{init}]$ and the observed point cloud $\mathcal{O}$ of an object, our goal is to predict a relative transformation $[\mathbf{R}_\Delta|\mathbf{t}_\Delta|\mathbf{s}_\Delta]$ between the initial prediction and ground truth $[\mathbf{R}_{gt}|\mathbf{t}_{gt}|\mathbf{s}_{gt}]$ leveraging a shape prior $\mathcal{P}$, as shown in Eq. (1).

$$[\mathbf{R}_\Delta|\mathbf{t}_\Delta|\mathbf{s}_\Delta] = CATRE([\mathbf{R}_{init}|\mathbf{t}_{init}|\mathbf{s}_{init}], \mathcal{O}, \mathcal{P}). \tag{1}$$

Thereby, $\mathbf{R} \in SO(3)$, $\mathbf{t} \in \mathbb{R}^3$ and $\mathbf{s} \in \mathbb{R}^3$ respectively represents rotation, translation and size. The shape prior $\mathcal{P}$ can be the mean shape of a given category or a generic skeleton like bounding box corners. We use the categorical mean shape ($c.f.$ Sec. 3.2) as the prior information in our main experiments and illustrations.

3.2 Overview of CATRE

Recapping Fig. 1, we first employ an off-the-shelf instance segmentation network ($e.g.$, Mask R-CNN [17]) to cut the object of interest from the observed depth

map. Then we back-project the depth into camera space and sample N_o points within a ball near to the object following [8,60]. The center and radius of this ball are determined by the initial pose estimate. To this end, we effectively remove outliers and obtain the observed point cloud $\mathcal{O} = \{o_i \in \mathbb{R}^3\}_{i=1}^{N_o}$.

To estimate the pose and size transformations based on the observed point cloud, we leverage the publicly available shape prior $\mathcal{P} = \{p_j \in \mathbb{R}^3\}_{j=1}^{N_p}$ from [47]. Thereby, a mean shape is reconstructed from the mean latent embedding for each category trained with a large amount of synthetic point cloud models from ShapeNet [5] using an encoder-decoder framework. We normalize the size of the shape prior such that the length of each side of its bounding box is 1.

Taking the observed point cloud $\mathcal{O}$, and the prior point cloud $\mathcal{P}$ as input, our CATRE network (Fig. 2) essentially predicts the relative transformation to iteratively refine the initial estimate towards the target pose and size.

3.3 Disentangled Transformation Learning

Point Clouds Focalization As the localization of input point clouds can vary dramatically in camera space, it would be very hard for the network to exploit useful information for learning relative transformation. Inspired by the image-space zooming in operation in DeepIM [30], we propose a novel point cloud focalization strategy to limit the range of input point clouds. Specifically, after transforming the prior point cloud $\mathcal{P}$ into camera space with the initial pose and size, we simultaneously shift the observed and estimated point clouds using the initial translation. Therefore, the input observed point cloud $\hat{\mathcal{O}}$ and prior point cloud $\hat{\mathcal{P}}$ can be obtained as follows

$$
\begin{aligned}
\hat{\mathcal{O}} &= \{\hat{o}_i \mid \hat{o}_i = o_i - \mathbf{t}_{init}\}_{i=1}^{N_o}, \\
\hat{\mathcal{P}} &= \{\hat{p}_j \mid \hat{p}_j = \mathbf{s}_{init} \odot \mathbf{R}_{init} p_j\}_{j=1}^{N_p},
\end{aligned}
\tag{2}
$$

where $\odot$ denotes element-wise product. Geometrically, $\hat{\mathcal{P}}$ and $\hat{\mathcal{O}}$ contain all the information needed for learning the relative transformation between the initial estimate and the target pose, no matter how we shift them. By fixing the center of $\hat{\mathcal{P}}$ to the origin, we can let the network focalize on the limited range around the origin, thus reducing the difficulty of learning for the network. Notice that, compared with similar strategies in image space proposed for instance-level 6D pose refinement and tracking [30,59], which rely on rendering with known CAD models and the costly operation of cropping and resizing, our point cloud focalization is much simpler yet more efficient.

Shared Encoder Inspired by works on learning-based point cloud registration [28,58,43], we employ a shared encoder to respectively extract embeddings $F_p \in \mathbb{R}^{N_p \times d}$ and $F_o \in \mathbb{R}^{N_o \times d}$ from $\hat{\mathcal{P}}$ and $\hat{\mathcal{O}}$ for efficiently encoding the geometric information locally and globally. Here $d = 1088$ denotes the output dimension. In point cloud registration, both point-based encoder [40] and graph-based encoder [33] have been widely used. However, we experimentally find 3D-GCN [33]

performing poorly in our work, as it is locally sensitive whilst the local graph structures of the partially observed point cloud and the full shape prior share little similarity. Hence we choose the simpler PointNet [40] as the backbone for better generalizability. Concretely, PointNet maps each point from $\mathbb{R}^3$ into a high-dimensional space, and generates an additional global feature by aggregating features from all points. The global feature is repeated N times and appended to each point-wise feature, where N denotes the number of input points. For details of the employed encoder, please kindly refer to the supplementary material.

By the prominent weight sharing strategy, we map $\hat{\mathcal{P}}$ and $\hat{\mathcal{O}}$ into a common feature space while remaining sensitive to relevant features for rigid transformation. Considering that rotation is different from translation and size in essence, we introduce a disentangled transformation learning strategy for them by means of two different heads namely *Rot-Head* and *TS-Head* in the following.

Rotation Prediction As aforementioned, $\hat{\mathcal{O}}$ encodes the information of $\mathbf{R}_{gt}$, and $\hat{\mathcal{P}}$ contains $\mathbf{R}_{init}$, so the network can predict $\mathbf{R}_\Delta$ by comparing $\hat{\mathcal{O}}$ and $\hat{\mathcal{P}}$. First, we concatenate F_o and F_p along the point dimension to get a unified per-point feature $F_{op} \in \mathbb{R}^{(N_o+N_p) \times d}$. Then, the feature is fed to Rot-Head to predict a continuous 6-dimensional rotation representation $[\mathbf{r}_\Delta^x | \mathbf{r}_\Delta^y]$ following [66,9], *i.e.*, the first two columns of the rotation matrix $\mathbf{R}_\Delta$. Finally, the relative rotation matrix $\mathbf{R}_\Delta$ is recovered as in [66], and the refined rotation prediction can be computed by $\mathbf{R}_{est} = \mathbf{R}_\Delta \mathbf{R}_{init}$, where $\mathbf{R}_{est}$ is used for loss calculation and reassigned as $\mathbf{R}_{init}$ for the next iteration.

Noteworthy, since the prediction of relative rotation is sensitive to local geometry, it is crucial how the network fuses the information of F_{op} at point level. Instead of using non-parametric operations like max-pooling which loses the local information severely, we opt for a trainable way to aggregate the point-wise features in Rot-Head. In particular, we employ two parallel branches for Rot-Head to obtain axis-wise rotation predictions $\mathbf{r}_\Delta^x$ and $\mathbf{r}_\Delta^y$, respectively. Each branch is comprised of two 1D convolutional layers each followed by Group Normalization [61] and GELU [18] activation to generate an implicit per-point rotation prediction $f_{op} \in \mathbb{R}^{(N_o+N_p) \times 3}$ first. Then the explicit global axis-wise prediction is aggregated from the permuted point-wise implicit prediction $f_{op}' \in \mathbb{R}^{3 \times (N_o+N_p)}$ using an extra 1D convolutional layer, reducing the dimension from $3 \times (N_o+N_p)$ to 3×1. In this way, the network can preserve the local information regarding the relative rotation between $\hat{\mathcal{P}}$ and $\hat{\mathcal{O}}$ in a trainable way.

Translation and Size Prediction Since $\mathcal{O}$ is shifted to $\hat{\mathcal{O}}$ by $\mathbf{t}_{init}$, the translation transformation $\mathbf{t}_\Delta$ can be derived from F_o directly. Different from predicting rotation by aggregating point-level information, we argue that translation and size should be treated differently in that they are both global attributions at the object level. Therefore, TS-Head first extracts a global feature f_o from F_o by performing max-pooling along the point dimension. Then, the initial scale is explicitly appended to f_o for predicting the size transformation $\mathbf{s}_\Delta$. Afterwards, two fully connected (FC) layers are applied to the global feature, reducing the

dimension from 1091 to 256. Finally, two parallel FC layers output $\mathbf{t}_\Delta$ and $\mathbf{s}_\Delta$ separately. The predicted translation and size are obtained by $\mathbf{t}_{est} = \mathbf{t}_{init} + \mathbf{t}_\Delta$ and $\mathbf{s}_{est} = \mathbf{s}_{init} + \mathbf{s}_\Delta$ accordingly. Similar to the rotation prediction, they are also re-assigned as the initial estimates for the next iteration.

3.4   Training and Testing Protocol

Pose Loss In works for direct pose regression, the design of loss function is crucial for optimization [54]. Apart from using an angular distance based loss on rotation [23], and L_1 losses for translation and scale, we additionally leverage a novel loss modified based on point matching [30] to couple the estimation of $[\mathbf{R}|\mathbf{t}|\mathbf{s}]$. To summarize, the overall loss function can be written as

$$\mathcal{L} = \mathcal{L}_{pm} + \mathcal{L}_{\mathbf{R}} + \mathcal{L}_{\mathbf{t}} + \mathcal{L}_{\mathbf{s}}, \tag{3}$$

where

$$\begin{cases} \mathcal{L}_{pm} & = \underset{\mathbf{x}\in\mathcal{P}}{\mathrm{avg}}\|(\mathbf{s}_{gt}\odot\mathbf{R}_{gt}\mathbf{x}+\mathbf{t}_{gt}) - (\mathbf{s}_{est}\odot\mathbf{R}_{est}\mathbf{x}+\mathbf{t}_{est})\|_1, \\ \mathcal{L}_{\mathbf{R}} & = \frac{3-\mathrm{Tr}(\mathbf{R}_{gt}\mathbf{R}_{est}^T)}{4}, \\ \mathcal{L}_{\mathbf{t}} & = \|\mathbf{t}_{gt}-\mathbf{t}_{est}\|_1, \\ \mathcal{L}_{\mathbf{s}} & = \|\mathbf{s}_{gt}-\mathbf{s}_{est}\|_1. \end{cases} \tag{4}$$

Here $\odot$ denotes element-wise product. For symmetric objects, we rotate $\mathbf{R}_{gt}$ along the known symmetry axis to find the best match for $\mathbf{R}_{est}$.

Iterative Training and Testing We employ an iterative training and testing strategy for pose refinement inspired by [29], namely the current prediction is re-assigned as the initial pose for the next iteration. By repeating this procedure multiple times, *i.e.*, 4 in this work, not only the diversity of training pose error distribution is enriched, but also more accurate and robust inference results can be obtained. To decouple CATRE from different initial pose estimation methods, during training we add random Gaussian noise to $[\mathbf{R}_{gt}|\mathbf{t}_{gt}|\mathbf{s}_{gt}]$ as initial pose in the first iteration. The network can therefore refine the pose prediction from various methods [55,32,47] consistently without re-training (see also Table 2).

3.5   Pose Tracking

It is natural to apply CATRE to tracking due to the similarity of pose refinement and tracking. Given a live point cloud stream and the 9DoF pose estimate $[\mathbf{R}_t|\mathbf{t}_t|\mathbf{s}_t]$ of an object at frame t, the target of category-level pose tracking is to estimate the object pose $[\mathbf{R}_{t+1}|\mathbf{t}_{t+1}|\mathbf{s}_{t+1}]$ of the next frame $t+1$. Similar to pose refinement, we use $[\mathbf{R}_t|\mathbf{t}_t|\mathbf{s}_t]$ as the initial pose to focalize $\mathcal{P}$ and $\mathcal{O}$, and predict the relative pose transformation between frame t and $t+1$.

During inference, we jitter the ground-truth pose of the first frame with the same Gaussian noise distribution in [60]. The pose is re-initialized if the adjacent frames are not consecutive.

4   Experiments

4.1   Experimental Setup

Implementation Details All the experiments are implemented using PyTorch [37]. We train our model using Ranger optimizer [34,65,63] with a base learning rate of 1×10^{-4}, annealed at 72% of the training epoch using a cosine schedule [24]. The total training epoch is set to 120 with a batch size of 16. We employ some basic strategies for depth augmentation, *e.g.*, randomly dropping depth values, adding Gaussian noise, and randomly filling 0 points, as well as some pose augmentations like random scaling, random rotation and translation perturbation as in [9]. Unless otherwise specified, the size of input images are 480×640, N_o and N_p are empirically set to 1024, and a single model is trained for all categories.

Datasets We conduct experiments on 4 datasets: REAL275 [55], CAMERA25 [55], LM [19], and YCB-V [62]. *REAL275* is a widely-used real-world category-level pose benchmark containing 4.3k images of 7 scenes for training, and 2.75k images of 6 scenes for testing. *CAMERA25* is a synthetic dataset generated by a context-aware mixed reality approach. There are 300k composite images of 1085 object instances in total, where 25k images and 184 instances are used for evaluation. REAL275 and CAMERA25 share the same object categories, *i.e.*, bottle, bowl, camera, can, laptop, and mug. *LM* dataset consists of 13 object instances, each of which is labeled around the center of a sequence of ≈1.2k cluttered images. We follow [4] to split ≈15% images for training and ≈85% images for testing. Additionally, 1k rendered images for each instance are used to assist training as in [31]. *YCB-V* dataset is very challenging due to severe occlusions and various lighting conditions. We select two objects, *i.e.*, master chef can and cracker box, out of 21 instances for unseen category refinement.

Metrics On REAL275 and CAMERA25, we follow [55] to report the mean Average Precision (mAP) of intersection over union (IoU) metric at different thresholds[2], as well as mAP at $n° \, m \, cm$ for evaluating pose accuracy. Additionally, for the task of tracking, we report mIoU, $\mathbf{R}_{err}$, and $\mathbf{t}_{err}$ as in [60], respectively standing for the mean IoU of ground-truth and predicted bounding boxes, average rotation error, and average translation error. On LM, the ADD metric [19] is used to measure whether the average offset of transformed model points is less than 10% of the object diameter. For symmetric objects, the ADD-S metric [20] is employed. On YCB-V, we use the area under the accuracy curves (AUC) of ADD(-S) metric varying the threshold from 0 to 10 cm as in [62].
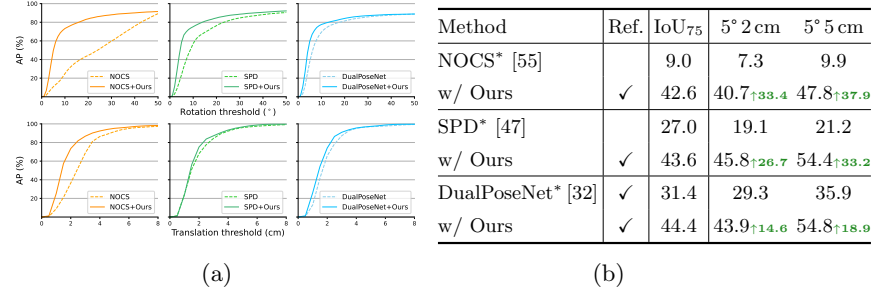
4.2   Category-level Pose Refinement

Results The quantitative results for category-level pose refinement on REAL275 and CAMERA25 are presented in Table 1. We use the results of SPD [47] as

---

[2]Note that there is a small mistake in the original IoU evaluation code of [55], we recalculated the IoU metrics as in [39].

Table 1: Results of category-level pose refinement on REAL275 and CAMERA25. SPD* denotes our re-implementation of SPD [47].

Dataset	Method	Refine	IoU_{75}	$5°\,2\,cm$	$5°\,5\,cm$	$10°\,2\,cm$	$10°\,5\,cm$
	NOCS [55]		9.4	7.2	10.0	13.8	25.2
	DualPoseNet [32]	✓	30.8	29.3	35.9	50	66.8
	CR-Net [56]		33.2	27.8	34.3	47.2	60.8
REAL275	SGPA [7]		37.1	35.9	39.6	61.3	70.7
	SPD [47]		27.0	19.3	21.4	43.2	54.1
	SPD*		27.0	19.1	21.2	43.5	54.0
	SPD*+Ours	✓	**43.6**	**45.8**	**54.4**	**61.4**	**73.1**
	NOCS [55]		37.0	32.3	40.9	48.2	64.6
	DualPoseNet [32]	✓	71.7	64.7	70.7	77.2	84.7
	CR-Net [56]		75.0	72.0	76.4	81.0	87.7
CAMERA25	SGPA [7]		69.1	70.7	74.5	82.7	88.4
	SPD [47]		46.9	54.3	59.0	73.3	81.5
	SPD*		46.9	54.1	58.8	73.9	82.1
	SPD*+Ours	✓	**76.1**	**75.4**	**80.3**	**83.3**	**89.3**

Table 2: Ablation on different initial poses on REAL275. *(a)* The average precision (AP) of rotation (°) and translation (cm) at different thresholds before and after refinement. *(b)* Quantitative results. * denotes our re-implementation.

Method	Ref.	IoU_{75}	$5°\,2\,cm$	$5°\,5\,cm$
NOCS* [55]		9.0	7.3	9.9
w/ Ours	✓	42.6	$40.7_{\uparrow33.4}$	$47.8_{\uparrow37.9}$
SPD* [47]		27.0	19.1	21.2
w/ Ours	✓	43.6	$45.8_{\uparrow26.7}$	$54.4_{\uparrow33.2}$
DualPoseNet* [32]	✓	31.4	29.3	35.9
w/ Ours	✓	44.4	$43.9_{\uparrow14.6}$	$54.8_{\uparrow18.9}$

(a)   (b)

initial poses during inference in our main experiments and ablation studies. It clearly shows that our method surpasses the baseline by a large margin, especially on the strict $5°\,2\,cm$ metric, achieving an absolute improvement of 26.7% on the challenging REAL275 dataset and 21.3% on the CAMERA25 dataset.

The results also demonstrate the significant superiority of CATRE over state-of-the-art methods [55,32,56,7,47]. Notice that although we do not exploit any color information in the refinement procedure, we are still very competitive overall. More importantly, on the strict metrics IoU_{75} and $5°\,2\,cm$, we surpass all the previous methods. We kindly refer readers to the *supplementary material* for more qualitative analyses and detailed results for each category.

Ablation Study Table 2 and Table 3 present the ablations on various factors.

Table 3: Ablation studies on REAL275. *(a)* Accuracy and speed w.r.t. iterations. *(b)* Quantitative results, where SPD* denotes our re-implementation of SPD [47].
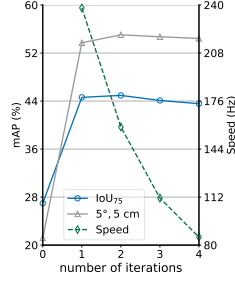
Row	Method	IoU_{75}	$5°\,2\,cm$	$5°\,5\,cm$	$5°$	$2\,cm$
A0	SPD*	27.0	19.1	21.2	23.8	68.6
B0	SPD*+Ours	43.6	**45.8**	**54.4**	58.0	75.1
C0	B0: Mean shape $\rightarrow$ 3D bounding box	42.9	39.5	47.2	52.7	73.1
C1	B0: Mean shape $\rightarrow$ 3D axes	42.2	36.7	43.1	49.6	73.2
D0	B0: TS-Head also w/ MaxPool(F_p)	**47.8**	45.7	52.8	**58.4**	75.7
D1	B0: w/o focalizing to origin	41.4	40.2	47.5	52.3	70.7
D2	B0: w/o adaptive points sampling	29.0	26.7	42.3	46.3	63.3
E0	B0: PointNet [40] $\rightarrow$ 3D-GCN [33]	28.4	36.0	43.4	47.7	68.0
E1	B0: w/o weight sharing	23.6	1.7	2.0	2.5	76.0
F0	B0: Single pose head (TS-Head style)	46.4	40.7	46.2	52.6	75.9
F1	B0: Single pose head (Rot-Head style)	45.0	41.1	50.0	55.1	69.1
F2	B0: Rot-Head conv fusion $\rightarrow$ MaxPool	45.1	41.8	49.5	55.0	76.4
F3	B0: Rot-Head conv fusion $\rightarrow$ AvgPool	44.9	41.8	48.0	53.4	**77.1**
F4	B0: TS-Head global feature$\rightarrow$point-wise	29.8	32.3	46.3	53.0	57.5
G0	B0: w/o $\mathcal{L}_{pm}$	44.4	42.5	50.7	55.5	74.1
G1	B0: w/o $\mathcal{L}_\mathbf{R}/\mathcal{L}_\mathbf{t}/\mathcal{L}_\mathbf{s}$	38.4	22.6	30.1	34.0	72.8

(a)   (b)

Ablation on Different Initial Poses To demonstrate the generalizability and robustness of our method w.r.t. different kinds of initial poses, we initialize the network with predictions of various qualities from [55,47,32] and present the results in Table 2. The network achieves consistent enhancement using the same weight, no matter initialized with the early proposed methods [55,47], or recent state of the art [32]. Table 2a further shows distinct improvement on rotation estimate after refinement.

Ablation on Shape Prior Aside from the mean shape, CATRE can use the eight corners of the 3D bounding box ($N_p = 8$) or four points constructing 3D axes ($N_p = 4$) as shape prior, the sizes of which are determined by $\mathbf{s}_{est}$. As shown in Table 3b, the results using mean shape (B0) are superior to those using bounding box corners (C0) or axes (C1). However, the poses can still be refined taking bounding box corners or axes as input, which implies that the geometric encoder is able to encode the initial pose information from sparse input without categorical prior knowledge. Inspired by this, we further extend our method to pose refinement on the unseen category in Sec. 4.5.

Effectiveness of Point Cloud Focalization To verify the efficiency and accuracy of predicting relative pose from the focalized point clouds, we conduct several experiments. Firstly, the input of TS-Head is replaced with a concatenated feature from both MaxPool(F_o) and MaxPool(F_p). Since the relative translation can be derived from $\hat{\mathcal{O}}$ due to the focalization operation, adding feature from $\hat{\mathcal{P}}$ introduces unnecessary noise thus leading to a slight performance drop (Table 3b B0 *vs.* D0). Moreover, we translate the center of shape prior $\mathcal{P}$ to $\mathbf{t}_{init}$,

and keep the pose of the observed point cloud $\mathcal{O}$ as it is in camera space. The initial translation is then appended to the input of TS-Head for better predicting $\mathbf{t}_\Delta$. However, we observe a larger decrease in accuracy (Table 3b B0 *vs.* D1) discarding focalization, thus limiting the absolute range of input point clouds is vital for reducing the complexity of learning relative transformation. To avoid introducing too many background points, we employed an adaptive strategy to sample points within a ball determined by the initial pose estimate. Without this step, the accuracy decreased dramatically (Table 3b B0 *vs.* D2).

Effectiveness of Shared Encoder As mentioned in Sec. 3.3, 3D-GCN [33] performs worse than PointNet [40] in our work (Table 3b B0 *vs.* E0). Moreover, weight sharing in the encoder is prominent for capturing relevant features regarding relative transformation by embedding $\hat{\mathcal{P}}$ and $\hat{\mathcal{O}}$ into a unified space, which is especially crucial to the prediction of $\mathbf{R}_\Delta$. Otherwise, it is almost unable to perform any refinement on rotation, as shown in Table 3b (B0 *vs.* E1).

Effectiveness of Disentangled Heads We consider two types of single pose heads, *i.e.*, extending TS-Head to estimate $[\mathbf{r}_\Delta^x | \mathbf{r}_\Delta^y]$ (F0), and adding two FC layers in Rot-Head to predict $\mathbf{t}_\Delta$ and $\mathbf{s}_\Delta$ (F1). Table 3b reveals that the disentangled design (B0) outperforms a single pose head leveraging a global feature (F0) or point-wise features (F1) consistently.

The Design Choices of Heads When replacing the trainable point-wise features aggregation with non-parametric operation MaxPool or AvgPool in Rot-Head, the accuracy on 5° drops significantly (Table 3b B0 *vs.* F2, F3). Furthermore, treating $\mathbf{t}$ and $\mathbf{s}$ as global attributes rather than collecting point-wise features like Rot-Head also enhances performance (Table 3b B0 *vs.* F4).

Effectiveness of Loss Function Table 3b (B0 *vs.* G0) evinces that $\mathcal{L}_{pm}$ coupling pose prediction brings performance enhancement, and the individual loss components designated for $[\mathbf{R}_{est} | \mathbf{t}_{est} | \mathbf{s}_{est}]$ are crucial as well (Table 3b B0 *vs.* G1).

Accuracy and Speed vs. Iteration In Table 3a, we show the accuracy and inference speed w.r.t. iterations. Here we empirically set 4 as the maximum number of iteration to balance the performance and speed. On a machine with a TITAN XP GPU and two Intel 2.0 GHz CPUs, CATRE runs at ≈ 85.32 Hz for 4 iterations.

4.3   Category-level Pose Tracking

Table 4 summarizes the quantitative results for category-level pose tracking, with a comparison between our methods and other tracking-based methods [51,12,60].

Leveraging a single model for tracking all objects, our method achieves competitive results with previous state-of-the-art methods. Noticeably, the pipeline can run at 89.21 Hz, which is 7x faster than the state-of-the-art method CAPTRA [60] and qualified for real-time application. Moreover, our method exceeds CAPTRA in the metric of mIoU and $\mathbf{t}_{err}$ by a large margin, in that CAPTRA

Table 4: Results of category-level pose tracking on REAL275. Following [60], we perturb the ground-truth pose of the first frame as the initial pose.

Method	Oracle ICP [60]	6-PACK [51]	iCaps [12]	CAPTRA [60]	Ours
Init.	GT.	GT. Pert.	Det. and Seg.	GT. Pert.	GT. Pert.
Speed (Hz) $\uparrow$	-	10	1.84	12.66	**89.21**
mIoU $\uparrow$	14.7	-	-	64.1	**80.2**
5° 5 cm$\uparrow$	0.7	33.3	31.6	**62.2**	57.2
$\mathbf{R}_{err}$(°) $\downarrow$	40.3	16.0	9.5	**5.9**	6.8
$\mathbf{t}_{err}$(cm) $\downarrow$	7.7	3.5	2.3	7.9	**1.2**

Table 5: Results on LM referring to the Average Recall (%) of ADD(-S) metric. * denotes symmetric objects. Ours(B) and Ours(F) use 8 bounding box corners and 128 FPS model points as shape priors respectively.

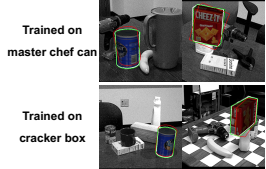| Method	Ref.	Ape	Bvise	Cam	Can	Cat	Drill	Duck	Ebox*	Glue*	Holep	Iron	Lamp	Phone	Mean
SSD-6D [26]+ICP	✓	65	80	78	86	70	73	66	**100**	**100**	49	78	73	79	79
DenseFusion [52]	✓	92.3	93.2	94.4	93.1	96.5	87.0	92.3	99.8	**100**	92.1	97	95.3	92.8	94.3
CloudAAE [15]+ICP	✓	-	-	-	-	-	-	-	-	-	-	-	-	-	95.5
PoseCNN [62]		27.9	69.4	47.7	71.1	56.4	65.4	43.2	98.2	95.0	50.7	65.9	70.4	54.4	62.7
w/ Ours(B)	✓	63.7	98.6	89.7	96.1	84.3	98.6	63.9	99.8	99.4	93.2	98.4	98.7	97.5	90.9
w/ Ours(F)	✓	**94.1**	**99.5**	**97.9**	**99.4**	**98.0**	**99.8**	**92.9**	99.8	99.7	**98.3**	**99.0**	**99.8**	**99.5**	**98.3**

solves $\mathbf{t}$ and $\mathbf{s}$ using the indirect Umeyama algorithm without RANSAC, making their method sensitive to outliers. In contrast, our network predicts $\mathbf{t}$ and $\mathbf{s}$ in an end-to-end manner. Additionally, our method deals with all the objects by a single network, while [60] needs to train separately for $\mathbf{R}$ and $\mathbf{t}/\mathbf{s}$ of each category. To sum up, although CAPTRA performs slightly better than ours w.r.t. 5° 5 cm and $\mathbf{R}_{err}$, we still have an edge over them in tracking speed and translation accuracy, which are vital in many robotic tasks such as grasping.

4.4 Instance-level Pose Refinement

Different from category-level refinement, instance-level pose refinement assumes the CAD model is available. Considering this, we propose two types of shape priors, *i.e.*, the tight bounding box of CAD model or N_p model points selected by farthest point sampling (FPS) algorithm, where we choose $N_p = 128$ empirically.

Table 5 shows the experimental results on LM dataset. Our methods achieve significant improvement against the initial prediction provided by PoseCNN [62]. Noteworthy, Ours(B) utilizing the sparse bounding box corners information can still be comparable with methods refined by ICP procedure leveraging exact CAD models. Furthermore, by explicitly exploiting the shape information with 128 FPS model points, Ours(F) achieves state of the art distinctly surpassing previous RGB-D based methods [26,52,15] with refinement.

Table 6: Results of unseen category pose refinement on YCB-Video. *(a)* Qualitative results, where white, red and green contours illustrate the ground-truth, initial (modified from PoseCNN [62]) and refined (Ours) poses, respectively. *(b)* Quantitative results (Metric: AUC of ADD(-S)).

Test Train	Master Chef Can		Cracker Box	
	Init.	Ref.	Init.	Ref.
Master Chef Can	52.7	71.3	57.8	65.3$_{\uparrow 7.5}$
Cracker Box	52.7	57.4$_{\uparrow 4.7}$	57.8	89.6

(a)   (b)

4.5 Pose Refinement on Unseen Category

As discussed before, utilizing bounding box corners as shape prior offers a possibility for pose refinement on unseen category, as long as the objects have a unified definition of the canonical view. We verify this assumption in two instances on the challenging YCB-V dataset, *i.e.*, master chef can and cracker box, and present qualitative and quantitative results in Table 6. Training with master chef can, the network is able to refine the pose of the unseen category cracker box and vice versa. It demonstrates that the network learns some general features for pose refinement exceeding the limitation of the trained category. We hope this experiment will inspire future work for unseen object refinement or tracking leveraging self-supervision or domain adaptation techniques.

5 Conclusion

We have presented CATRE, a versatile approach for pose refinement and tracking, which enables tackling instance-level, category-level and unseen category problems with a unified framework. The key idea is to iteratively align the focalized shape prior and observed point cloud by predicting a relative pose transformation using disentangled pose heads. Our algorithm is generalizable to various kinds of baselines and achieves significant boosts in performance over other state-of-the-art category-level pose estimation works as well as competitive results with tracking-based methods. Future work will focus on more challenging scenarios, such as tracking unseen objects, leveraging RGB information, and training with only synthetic data utilizing large-scale 3D model sets [5,11,41].

References

1. Aoki, Y., Goforth, H., Srivatsan, R.A., Lucey, S.: PointNetLK: Robust & efficient point cloud registration using PointNet. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 7163–7172 (2019)
2. Besl, P.J., McKay, N.D.: A method for registration of 3-D shapes. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) **14**(2), 239–256 (1992)
3. Bouaziz, S., Tagliasacchi, A., Pauly, M.: Sparse iterative closest point. In: Computer graphics forum. vol. 32, pp. 113–123. Wiley Online Library (2013)
4. Brachmann, E., Michel, F., Krull, A., Ying Yang, M., Gumhold, S., Rother, C.: Uncertainty-driven 6D pose estimation of objects and scenes from a single RGB image. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 3364–3372 (2016)
5. Chang, A.X., Funkhouser, T., Guibas, L., Hanrahan, P., Huang, Q., Li, Z., Savarese, S., Savva, M., Song, S., Su, H., et al.: ShapeNet: An information-rich 3D model repository. arXiv preprint arXiv:1512.03012 (2015)
6. Chen, D., Li, J., Wang, Z., Xu, K.: Learning canonical shape space for category-level 6D object pose and size estimation. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 11970–11979 (2020). https://doi.org/10.1109/CVPR42600.2020.01199
7. Chen, K., Dou, Q.: SGPA: Structure-guided prior adaptation for category-level 6D object pose estimation. In: IEEE/CVF International Conference on Computer Vision (ICCV). pp. 2773–2782 (2021)
8. Chen, W., Jia, X., Chang, H.J., Duan, J., Leonardis, A.: G2L-Net: Global to local network for real-time 6D pose estimation with embedding vector features. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 4233–4242 (2020)
9. Chen, W., Jia, X., Chang, H.J., Duan, J., Linlin, S., Leonardis, A.: FS-Net: Fast shape-based network for category-level 6D object pose estimation with decoupled rotation mechanism. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1581–1590 (June 2021)
10. Choy, C., Dong, W., Koltun, V.: Deep global registration. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 2514–2523 (2020)
11. Collins, J., Goel, S., Luthra, A., Xu, L., Deng, K., Zhang, X., Yago Vicente, T.F., Arora, H., Dideriksen, T., Guillaumin, M., Malik, J.: ABO: Dataset and benchmarks for real-world 3d object understanding. arXiv preprint arXiv:2110.06199 (2021)
12. Deng, X., Geng, J., Bretl, T., Xiang, Y., Fox, D.: iCaps: Iterative category-level object pose and shape estimation. IEEE Robotics and Automation Letters (RAL) (2022)
13. Du, G., Wang, K., Lian, S., Zhao, K.: Vision-based robotic grasping from object localization, object pose estimation to grasp estimation for parallel grippers: a review. Artificial Intelligence Review **54**(3), 1677–1734 (2021)
14. Fan, Z., Song, Z., Xu, J., Wang, Z., Wu, K., Liu, H., He, J.: ACR-Pose: Adversarial canonical representation reconstruction network for category level 6d object pose estimation. arXiv preprint arXiv:2111.10524 (2021)
15. Gao, G., Lauri, M., Hu, X., Zhang, J., Frintrop, S.: CloudAAE: Learning 6D object pose regression with on-line data synthesis on point clouds. In: IEEE International Conference on Robotics and Automation (ICRA). pp. 11081–11087 (2021)

16. Groueix, T., Fisher, M., Kim, V.G., Russell, B.C., Aubry, M.: 3D-CODED: 3D correspondences by deep deformation. In: European Conference on Computer Vision (ECCV) (September 2018)
17. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask R-CNN. In: IEEE/CVF International Conference on Computer Vision (ICCV). pp. 2961–2969 (2017)
18. Hendrycks, D., Gimpel, K.: Gaussian error linear units (gelus). arXiv preprint arXiv:1606.08415 (2016)
19. Hinterstoisser, S., Lepetit, V., Ilic, S., Holzer, S., Bradski, G., Konolige, K., , Navab, N.: Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes. In: Asian Conference on Computer Vision (ACCV) (2012)
20. Hodaň, T., Matas, J., Obdržálek, Š.: On evaluation of 6d object pose estimation. European Conference on Computer Vision Workshops (ECCVW) pp. 606–619 (2016)
21. Hodan, T., Sundermeyer, M., Drost, B., Labbe, Y., Brachmann, E., Michel, F., Rother, C., Matas, J.: BOP Challenge 2020 on 6D object localization. In: Bartoli, A., Fusiello, A. (eds.) European Conference on Computer Vision Workshops (ECCVW). pp. 577–594 (2020)
22. Huang, S., Qi, S., Xiao, Y., Zhu, Y., Wu, Y.N., Zhu, S.C.: Cooperative holistic scene understanding: Unifying 3d object, layout, and camera pose estimation. Advances in Neural Information Processing Systems (NeurIPS) **31** (2018)
23. Huynh, D.Q.: Metrics for 3D rotations: Comparison and analysis. Journal of Mathematical Imaging and Vision **35**(2), 155–164 (2009)
24. Ilya Loshchilov, F.H.: SGDR: stochastic gradient descent with warm restarts. In: International Conference on Learning Representations (ICLR) (2017)
25. Iwase, S., Liu, X., Khirodkar, R., Yokota, R., Kitani, K.M.: RePOSE: Fast 6D object pose refinement via deep texture rendering. In: IEEE/CVF International Conference on Computer Vision (ICCV). pp. 3303–3312 (2021)
26. Kehl, W., Manhardt, F., Tombari, F., Ilic, S., Navab, N.: SSD-6D: Making RGB-based 3D detection and 6D pose estimation great again. In: IEEE/CVF International Conference on Computer Vision (ICCV). pp. 1521–1529 (2017)
27. Labbé, Y., Carpentier, J., Aubry, M., Sivic, J.: CosyPose: Consistent multi-view multi-object 6D pose estimation. In: European Conference on Computer Vision (ECCV). pp. 574–591 (2020)
28. Lee, D., Hamsici, O.C., Feng, S., Sharma, P., Gernoth, T.: DeepPRO: Deep partial point cloud registration of objects. In: IEEE/CVF International Conference on Computer Vision (ICCV). pp. 5683–5692 (2021)
29. Li, Y., Wang, G., Ji, X., Xiang, Y., Fox, D.: DeepIM: Deep iterative matching for 6D pose estimation. In: European Conference on Computer Vision (ECCV). pp. 683–698 (2018)
30. Li, Y., Wang, G., Ji, X., Xiang, Y., Fox, D.: DeepIM: Deep iterative matching for 6D pose estimation. International Journal of Computer Vision (IJCV) **128**(3), 657–678 (2020)
31. Li, Z., Wang, G., Ji, X.: CDPN: Coordinates-Based Disentangled Pose Network for Real-Time RGB-Based 6-DoF Object Pose Estimation. In: IEEE/CVF International Conference on Computer Vision (ICCV). pp. 7678–7687 (2019)
32. Lin, J., Wei, Z., Li, Z., Xu, S., Jia, K., Li, Y.: DualPoseNet: Category-level 6D object pose and size estimation using dual pose network with refined learning of pose consistency. In: IEEE/CVF International Conference on Computer Vision (ICCV). pp. 3560–3569 (October 2021)

33. Lin, Z.H., Huang, S.Y., Wang, Y.C.F.: Convolution in the cloud: Learning deformable kernels in 3D graph convolution networks for point cloud analysis. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1800–1809 (2020)

34. Liu, L., Jiang, H., He, P., Chen, W., Liu, X., Gao, J., Han, J.: On the variance of the adaptive learning rate and beyond. In: International Conference on Learning Representations (ICLR) (2019)

35. Marchand, E., Uchiyama, H., Spindler, F.: Pose estimation for augmented reality: a hands-on survey. IEEE Transactions on Visualization and Computer Graphics (TVCG) **22**(12), 2633–2651 (2015)

36. Nie, Y., Han, X., Guo, S., Zheng, Y., Chang, J., Zhang, J.J.: Total3DUnderstanding: Joint layout, object pose and mesh reconstruction for indoor scenes from a single image. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 55–64 (2020)

37. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al.: PyTorch: An imperative style, high-performance deep learning library. In: Advances in Neural Information Processing Systems (NeurIPS). pp. 8026–8037 (2019)

38. Peng, S., Liu, Y., Huang, Q., Zhou, X., Bao, H.: Pvnet: Pixel-wise voting network for 6dof pose estimation. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 4561–4570 (2019)

39. Peng, W., Yan, J., Wen, H., Sun, Y.: Self-supervised category-level 6D object pose estimation with deep implicit shape representation. Proceedings of the AAAI Conference on Artificial Intelligence (AAAI) **36**(2), 2082–2090 (Jun 2022). https://doi.org/10.1609/aaai.v36i2.20104

40. Qi, C.R., Su, H., Mo, K., Guibas, L.J.: PointNet: Deep learning on point sets for 3D classification and segmentation. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) **1**(2), 4 (2017)

41. Reizenstein, J., Shapovalov, R., Henzler, P., Sbordone, L., Labatut, P., Novotny, D.: Common objects in 3D: Large-scale learning and evaluation of real-life 3D category reconstruction. In: IEEE/CVF International Conference on Computer Vision (ICCV) (2021)

42. Rusinkiewicz, S., Levoy, M.: Efficient variants of the ICP algorithm. In: Proceedings third international conference on 3-D digital imaging and modeling. pp. 145–152. IEEE (2001)

43. Sarode, V., Li, X., Goforth, H., Aoki, Y., Srivatsan, R.A., Lucey, S., Choset, H.: Pcrnet: Point cloud registration network using pointnet encoding. arXiv preprint arXiv:1908.07906 (2019)

44. Segal, A., Haehnel, D., Thrun, S.: Generalized-ICP. In: Robotics: science and systems. vol. 2, p. 435. Seattle, WA (2009)

45. Song, C., Song, J., Huang, Q.: HybridPose: 6D object pose estimation under hybrid representations. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 431–440 (2020)

46. Su, Y., Rambach, J., Minaskan, N., Lesur, P., Pagani, A., Stricker, D.: Deep multi-state object pose estimation for augmented reality assembly. In: 2019 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct). pp. 222–227 (2019)

47. Tian, M., Ang Jr, M.H., Lee, G.H.: Shape prior deformation for categorical 6D object pose and size estimation. In: European Conference on Computer Vision (ECCV) (August 2020)

48. Trappolini, G., Cosmo, L., Moschella, L., Marin, R., Melzi, S., Rodolà, E.: Shape registration in the time of transformers. In: Advances in Neural Information Processing Systems (NeurIPS). vol. 34, pp. 5731–5744 (2021)
49. Tremblay, J., To, T., Sundaralingam, B., Xiang, Y., Fox, D., Birchfield, S.: Deep object pose estimation for semantic robotic grasping of household objects. In: Conference on Robot Learning (CoRL). pp. 306–316 (2018)
50. Umeyama, S.: Least-squares estimation of transformation parameters between two point patterns. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) **13**(04), 376–380 (apr 1991). https://doi.org/10.1109/34.88573
51. Wang, C., Martín-Martín, R., Xu, D., Lv, J., Lu, C., Fei-Fei, L., Savarese, S., Zhu, Y.: 6-PACK: Category-level 6D pose tracker with anchor-based keypoints. In: IEEE International Conference on Robotics and Automation (ICRA). pp. 10059–10066 (2020)
52. Wang, C., Xu, D., Zhu, Y., Martín-Martín, R., Lu, C., Fei-Fei, L., Savarese, S.: DenseFusion: 6D object pose estimation by iterative dense fusion. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 3343–3352 (2019)
53. Wang, G., Manhardt, F., Liu, X., Ji, X., Tombari, F.: Occlusion-aware self-supervised monocular 6D object pose estimation. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) (2021). https://doi.org/10.1109/TPAMI.2021.3136301
54. Wang, G., Manhardt, F., Tombari, F., Ji, X.: GDR-Net: Geometry-guided direct regression network for monocular 6D object pose estimation. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 16611–16621 (2021)
55. Wang, H., Sridhar, S., Huang, J., Valentin, J., Song, S., Guibas, L.J.: Normalized object coordinate space for category-level 6D object pose and size estimation. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 2642–2651 (2019)
56. Wang, J., Chen, K., Dou, Q.: Category-level 6D object pose estimation via cascaded relation and recurrent reconstruction networks. In: IEEE/RJS International Conference on Intelligent Robots and Systems (IROS) (2021)
57. Wang, Y., Solomon, J.: PRNet: self-supervised learning for partial-to-partial registration. In: Advances in Neural Information Processing Systems (NeurIPS). pp. 8814–8826 (2019)
58. Wang, Y., Solomon, J.M.: Deep closest point: Learning representations for point cloud registration. In: IEEE/CVF International Conference on Computer Vision (ICCV). pp. 3523–3532 (2019)
59. Wen, B., Mitash, C., Ren, B., Bekris, K.E.: se(3)-TrackNet: Data-driven 6D pose tracking by calibrating image residuals in synthetic domains. In: IEEE/RJS International Conference on Intelligent Robots and Systems (IROS). pp. 10367–10373 (2020)
60. Weng, Y., Wang, H., Zhou, Q., Qin, Y., Duan, Y., Fan, Q., Chen, B., Su, H., Guibas, L.J.: CAPTRA: Category-level pose tracking for rigid and articulated objects from point clouds. In: IEEE/CVF International Conference on Computer Vision (ICCV). pp. 13209–13218 (2021)
61. Wu, Y., He, K.: Group normalization. In: European Conference on Computer Vision (ECCV). pp. 3–19 (2018)
62. Xiang, Y., Schmidt, T., Narayanan, V., Fox, D.: PoseCNN: A convolutional neural network for 6D object pose estimation in cluttered scenes. Robotics: Science and Systems Conference (RSS) (2018)

63. Yong, H., Huang, J., Hua, X., Zhang, L.: Gradient centralization: A new optimization technique for deep neural networks. In: European Conference on Computer Vision (ECCV). pp. 635–652. Springer (2020)
64. Zakharov, S., Shugurov, I., Ilic, S.: DPOD: Dense 6D pose object detector in RGB images. In: IEEE/CVF International Conference on Computer Vision (ICCV) (2019)
65. Zhang, M., Lucas, J., Ba, J., Hinton, G.E.: Lookahead optimizer: k steps forward, 1 step back. In: Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., Garnett, R. (eds.) Advances in Neural Information Processing Systems (NeurIPS). vol. 32. Curran Associates, Inc. (2019)
66. Zhou, Y., Barnes, C., Lu, J., Yang, J., Li, H.: On the continuity of rotation representations in neural networks. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 5745–5753 (2019)