TTA-COPE: Test-Time Adaptation for Category-Level Object Pose Estimation

Taeyeop Lee[1] Jonathan Tremblay[2] Valts Blukis[2] Bowen Wen[2] Byeong-Uk Lee[1]
Inkyu Shin[1] Stan Birchfield[2] In So Kweon[1] Kuk-Jin Yoon[1]
[1]KAIST [2]NVIDIA

Abstract

Test-time adaptation methods have been gaining attention recently as a practical solution for addressing source-to-target domain gaps by gradually updating the model without requiring labels on the target data. In this paper, we propose a method of test-time adaptation for category-level object pose estimation called TTA-COPE. We design a pose ensemble approach with a self-training loss using pose-aware confidence. Unlike previous unsupervised domain adaptation methods for category-level object pose estimation, our approach processes the test data in a sequential, online manner, and it does not require access to the source domain at runtime. Extensive experimental results demonstrate that the proposed pose ensemble and the self-training loss improve category-level object pose performance during test time under both semi-supervised and unsupervised settings.

1. Introduction

Object pose estimation is a crucial problem in computer vision and robotics. Advanced methods that focus on diverse variations of object 6D pose estimation have been introduced, such as known 3D objects (instance-level) [28, 38], category-level [18, 36, 43], few-shot [52], and zero-shot pose estimation [13, 47]. These techniques are useful for downstream applications requiring an online operation, such as robotic manipulation [6, 25, 48] and augmented reality [23, 24, 32]. Our paper focuses on the category-level object pose estimation problem since it is more broadly applicable than the instance-level problem.

Many works on category-level object pose estimation [2, 3, 17, 18, 36, 43, 44] have been proposed recently. These approaches estimate multiple classes of object pose more efficiently in a single network compared to the instance-level object pose estimation methods [27, 38, 41, 49–51], which depend on known 3D shape knowledge and the size of the objects. Notably, Wang *et al*. [43] introduced a novel representation called Normalized Object Coordinate Space (NOCS) to align various object instances within each category in a canonical 3D space. The strengths of the NOCS representation have led to its adoption by follow-up work [3, 17, 36].

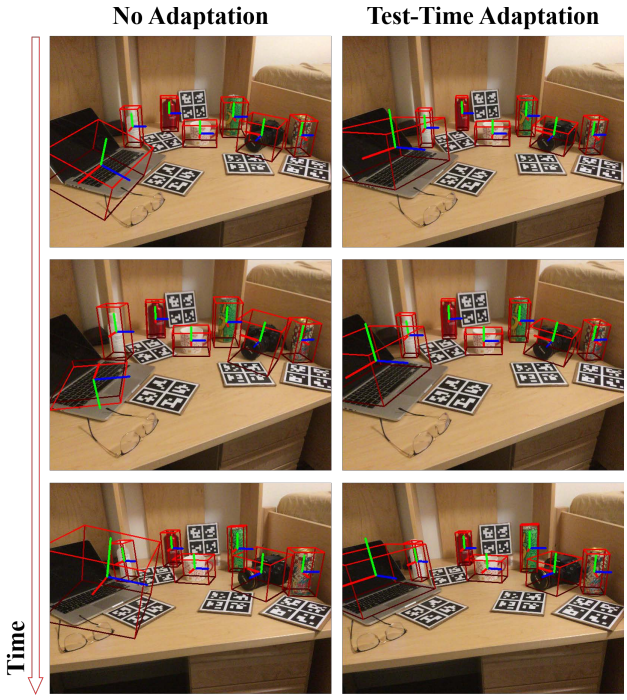

Figure 1. We propose a Test-Time Adaptation for Category-level Object Pose Estimation framework (TTA-COPE) that automatically improves the network in an online manner without labeled target data. As new image frames are processed, our method fine-tunes the network using the unlabeled data and simultaneously applies the network to perform pose estimation via inference. This approach successfully handles domain shifts compared with no adaptation, as seen here.

In order to obtain accurate category-level object pose methods in unseen real-world scenarios, it is desirable to fine-tune the models in the new environment with labeled target data. The model that is not fine-tuned on the target domain distribution will almost certainly exhibit lower performance than the fine-tuned model [37]. However, annotating 6D poses of objects in the target environment is an expensive process [1, 39, 43, 45] that we seek to avoid.

Table 1. **Comparison with prior unsupervised works for category-level object pose estimation.** Our unsupervised method trains models without 2D or 3D labels of target data, similar to Self-DPDN [16]. Unlike previous methods, our proposed approach updates the model online without offline batch processing. Moreover, we do not use the source data during test time (source-free) because it is impractical to train on a large amount of source data every iteration. There also may be privacy or legal constraints to access source data [21].

Method	Unsupervised		Test-time Adaptation	
	Target 3D	Target 2D	Source-Free	Online Adaptation
Supervised	✗	✗	✗	✗
SSC-6D [29]	✓	✗	✗	✗
RePoNet [5]	✓	✗	✗	✗
UDA-COPE [15]	✓	✗	✓	✗
Self-DPDN [16]	✓	✓	✗	✗
Ours	✓	✓	✓	✓

Compared to annotating in 2D space, labeling in 3D space requires specific knowledge about geometry [7] from the annotator and is much more laborious, time-consuming, and error-prone due to the complex nature of $SE(3)$ space. Therefore, it is usually challenging to annotate real-world data with 3D annotations for fine-tuning.

In order to solve the aforementioned problem of annotating object pose data in the real world, several recent methods [5, 15, 16] propose unsupervised domain adaptation (UDA) that aims to train the network without utilizing the ground truth of target pose labels. Although they show promising results using UDA techniques, these approaches still do not meet some of the requirements for online applications. For example, when a robot encounters a new environment, it is desirable to adapt the scene online manner while estimating object poses rather than waiting for enough data to be collected in the novel scene to train the model offline.

This problem definition of online fine-tuning is more practical for real applications, where we desire to update the model instantly when new data becomes available for fast domain adaptation. This setting is known as test-time adaptation (TTA) [42]. For TTA, the requirements are as follows: 1) labeled source data should not be accessed at test time, 2) adaptation should be online (rather than offline batch processing), and 3) the method should be fully unsupervised, without using 2D or 3D target labels during online fine-tuning. Since we do not have access to labeled source data (source-free) at test time this problem is more challenging than existing unsupervised category-level object pose methods [5,15,16,29]. Table 1 summarizes the difference between our problem definition and existing methods, showing that test-time adaptation for category-level object pose estimation remains an open problem.

In this paper, we propose Test-time Adaptation for Category-level Object Pose Estimation (**TTA-COPE**) to handle domain shifts without any target domain annotations (see Fig. 1). Prior works on general test-time adaptation [42,46] propose self-training to minimize entropy loss. TENT [42] has shown improvement in 2D classification and segmentation tasks. We show, however, that simply extending TENT for the category-level object pose estimation is not effective. Another self-training strategy is the teacher-student framework [35] with pseudo labels. However, since pseudo labels are created without any noise filtering, naive pseudo labels may be unreliable and cause convergence to a suboptimal model.

To tackle this problem, we design a novel pose ensemble method to perform test-time adaptation for category-level object pose estimation by extending the pose-aware filtering of UDA-COPE [15]. The proposed method uses an ensemble of teacher-student predictions based on pose-aware confidence, which is used both for generating pseudo labels and inference. Also, the pose ensemble helps to train models with additional self-training loss to reduce the domain shift for category-level pose estimation by using pose-aware confidence. We demonstrate the advantages of our proposed pose ensemble and self-training loss with extensive studies in both semi-supervised and unsupervised settings. We show that our TTA-COPE framework achieves state-of-the-art performance compared to strong TTA baselines.

In summary, the main contributions of our work are as follows:

- We propose Test-Time Adaptation for Category-level Object Pose Estimation (TTA-COPE), which handles domain shifts without labeling target data and without accessing source data during test time.
- We introduce a pose ensemble with self-training loss that utilizes the teacher-student predictions to generate robust pseudo labels and estimates accurate poses for inference.
- We evaluate our framework with experimental comparisons against strong test-time baselines and state-of-the-art methods under both semi-supervised and unsupervised settings.

2. Related Works

2.1. Supervised Methods

Fully supervised learning methods for category-level object pose estimation [2,3,17,36,43,44] train their models using labeled source data (*e.g.*, synthetic) and target data (*e.g.*, real). Most category-level 6D object pose and size estimation approaches [3, 14, 36, 43, 44] use the dense correspondence via Normalized Object Coordinate Space (NOCS) representation as a common way to estimate pose and size. These correspondence-based methods initially estimate the NOCS map from RGB or depth images. Afterwards, the Umeyama algorithm [40] with RANSAC is used to estimate

optimal poses and object sizes by minimizing distances between depth and estimated NOCS map.

Some methods use category priors [3,36,44] as the representative 3D shape per class, jointly reconstructing the full 3D shape and estimating the NOCS map from the full shape. Results show that this prior category helps improve the accuracy of the NOCS map and enhance the pose estimation performance. Other methods directly regress the pose or jointly utilize the correspondence representations [2,11,17].

2.2. Unsupervised Methods

Given that annotating the 6D object pose and 3D size labels in the real world is expensive, time-consuming, and laborious, RePoNet [5] and SSC-6D [29] propose semi-supervised approaches that reconstruct the entire shape and use differential rendering [19, 20] techniques for self-training signals. These are category-specific methods and utilize multiple models, as many models as the number of categories is required. UDA-COPE [15] proposes an unsupervised domain adaptation method to mitigate the domain shift from the source to the target domain with a single model to estimate all categories efficiently.

Although these works show reasonable performance without using pose labels, there is still a limitation in relying on 2D ground truth information (segmentation or bounding boxes) to train the segmentation network or pose network. Self-DPDN [16] shows a fully unsupervised method using inter/intra-consistency as a reconstructed shape in a self-supervised objective but requires a supervised loss for the source domain in the unsupervised learning process. Additionally, the self-training loss utilizes a full 3D shape and requires an additional reconstruction module for 3D shape. Most methods, except for UDA-COPE [15], jointly use the supervised loss using source label data during unsupervised learning to relax the unstable training. Furthermore, all the aforementioned unsupervised methods train their model in an offline manner and are unsuitable for online applications [24, 25].

2.3. Test-time Adaptation Methods

Test-Time Adaptation (TTA) aims to enable the online adaptation of a pretrained model to the target domain without access to the source domain (source-free) [26,34,42,46]. The source data is commonly inaccessible during inference time because it is inefficient to train on a huge amount of source data every iteration. There may also be privacy or legal constraints to accessing source data [21], thus TTA is a more difficult but practical task than UDA. From the point of view of real-world applications, it is necessary to adapt to the new scene in an online way. Accordingly, test-time adaptation is necessary for the success of practical, real-world computer vision applications. Wang *et al*. propose Test entropy minimization (TENT) [42], which trains a network using a labeled source dataset, and adapts it to the unlabeled target dataset by updating the network parameters in batch norm layers using entropy loss. CoTTA [46] proposes a continual test-time adaptation method on the 2D classification and semantic segmentation tasks, and it effectively reduces the error accumulations while continually changing target data. Not limited to the 2D tasks, test-time adaptation methods have been applied to other tasks such as 3D segmentation [31, 33] and robot manipulation [22].

3. TTA-COPE

Given an RGB-D image, our approach aims to estimate the 6D pose $T \in \mathrm{SE}(3)$ and size $s \in \mathbb{R}^3_+$ of each object. The object pose T is defined as the rigid transformation $[R \,|\, t]$, with rotation $R \in \mathrm{SO}(3)$ and translation $t \in \mathbb{R}^3$.

Our method consists of a two-stage learning scheme using source and target data, respectively. In the first stage, we train the network using labeled (synthetic) source data in a supervised manner (Sec. 3.1). This step is the same as supervised methods and results in a pretrained model. We then utilize this pretrained model for test-time adaptation (TTA) using unlabeled target data without accessing the source data. To the best of our knowledge, we are the first to propose test-time adaptation for category-level object pose estimation. Therefore, we study and explore how TTA baselines (Sec. 3.2) might apply to category-level object pose estimation. Finally, our proposed method is presented and explained (Sec. 3.3).

3.1. Pretraining with Source Data

In this section, we introduce an overview of our model and then describe how to train the model using labeled source data. We use UDA-COPE [15] network as a base model, which we supplement with batch normalization (BN) [10] in the 2D network to utilize BN updates for test-time adaptation. Fig. 2 shows an overview of our network, consisting of three steps to estimate the pose similar to correspondence methods [3, 16, 36]. First, we detect the bounding area of the object and segment the object surface using the off-the-shelf segmentation model from a single RGB image [8]. In the second stage, given a cropped 2D object image I and segmented 3D object point cloud D, our model estimates the correspondence as a NOCS map N. We leverage the NOCS representation [43] to align diverse object instances within each class in a unified 3D space. In the final stage, our proposed pose ensemble method estimates the object pose T. It simultaneously utilizes the predictions of the student and teacher model by calculating the pose confidence using the number of filtered points.

The network is pretrained in a fully supervised manner using labeled source data. We minimize the predictions of the NOCS map N using the cross-entropy loss L_{CE} with ground truth labels N^{GT}. We also use the consistency loss

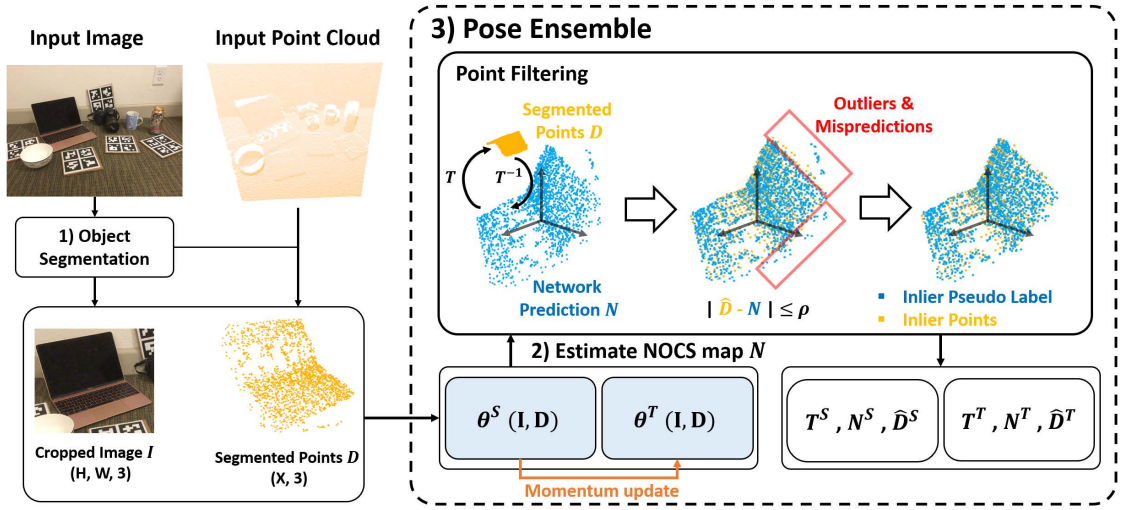

Figure 2. **Overview of our Test-Time Adaptation for Category-level Object Pose Estimation (TTA-COPE) framework.** Our method consists of three steps to estimate category-level object pose, similar to correspondence methods. 1) We detect and segment the object region using an off-the-shelf segmentation model from a single RGB image. 2) Given a cropped 2D object image I and segmented 3D object point cloud D, our model estimates the correspondence points as a NOCS map N. 3) Our proposed pose ensemble estimates the object pose T. It simultaneously utilizes the predictions of the student θ^S and teacher model θ^T by calculating the pose-aware confidence using the number of inlier points. We operate point filtering for each student and teacher model and select a confident pose T with higher confidence (*i.e.*, more inlier points) for which the pose estimation is more accurate.

L_C with 2D and 3D augmentation to make the network robust to noise in the input [15]. The supervised loss is formulated as:

$$L_{sup} = \lambda_{CE} L_{CE}(N, N^{GT}) + \lambda_C L_C(N_{aug}, N), \quad (1)$$

where N_{aug} is the estimated NOCS map from augmented 2D or 3D input, and $\lambda_{CE}, \lambda_C \in \mathbb{R}_+$ are weights.

3.2. Test-time Adaptation with Target Data

The model is updated every iteration while simultaneously estimating the object pose of the current scene. (Note that this is different from previous methods that update the model by running multiple epochs over the target data [16].)

We have a design choice for the objective loss when updating the model during test time. One of the widely used test-time objective losses is the entropy loss proposed by TENT [42],

$$L_{ent} = -\sum p(x_t) \log p(x_t), \quad (2)$$

where $p(x_t)$ is the probability of predictions $\theta^S(x_t)$ from target data x_t. This simple objective loss encourages sharp distributions by assigning the most probability.

Another common approach is utilizing pseudo labels from a teacher-student framework [35] with momentum update. The teacher model θ^T generates pseudo ground truth $\hat{y} = \theta^T(x_t)$, where $x_t = (I, D)$. The student model θ^S then uses the predictions of the teacher as the ground truth signals (pseudo labels) with cross-entropy loss,

$$L_{pl} = L_{CE}(\theta^S(x_t), \hat{y}). \quad (3)$$

After updating the student model $\theta_i^S \rightarrow \theta_{i+1}^S$ by minimizing Eq. (3), the teacher model θ_{i+1}^T is updated by momentum update,

$$\theta_{i+1}^T \leftarrow \gamma \theta_i^T + (1-\gamma)\theta_{i+1}^S, \quad (4)$$

where i stands for the time stamp of the iteration and γ is the momentum smoothing factor. The separation of the model structure mitigates some error accumulation by the pseudo label and momentum update.

Although these objective strategies from previous work potentially provide self-training signals for test-time adaptation, they become brittle when directly applied to a category-level object pose estimation task. The pseudo labels $\hat{y}$ are still inherently exposed to noisy inputs and predictions, leading to errors in the student model θ^S. Since θ^T cannot guarantee clean pseudo labels for the student, we need the ability to filter noise in the input or predictions and to yield more reliable pseudo labels to the θ^S.

3.3. Pose Ensemble

To solve this problem, we propose to extend the pseudo-label filtering proposed by UDA-COPE [15] with pose ensemble processing. We observe that the UDA-COPE framework and pose filtering, while showing notable results, is inefficient for two reasons. First, the teacher model is only used for generating pseudo labels and not for inference. Second, the student model is also only used for inference and has less influence in generating pseudo labels.

We propose instead to jointly utilize the teacher and student models for both generating pseudo labels and for infer-

ence, which we show in the experimental results produces a noticeable improvement. To this end, we design a pose ensemble module to simultaneously use teacher and student model predictions. Fig. 2 shows an overview of the pose ensemble with point filtering.

The point filtering has three steps. Unlike UDA-COPE [15], we repeat these steps for both the student and teacher: 1) We initially estimate T^S using the Umeyama algorithm θ_p within the region of object points D and estimated NOCS map N^S. 2) We transform observed depth to normalized object coordinate space $\widehat{D}^S$ by multiplying $(T^S)^{-1}$ and D. 3) We compute each matching point distance between N^S and $\widehat{D}$, removing outliers that exceed a certain threshold ρ. This whole process of point-filtering is then repeated using N^T to estimate T^T. This process is represented as follows, where $N^{S,T}$ is either N^S or N^T, and similarly for $T^{S,T}$:

$$\begin{aligned} T^{S,T} &= \theta_p(D, N^{S,T}), \\ \widehat{D}^{S,T} &= \left(T^{S,T}\right)^{-1} D, \\ e_j^{S,T} &= \begin{cases} inlier & \text{if } \|\widehat{D}_j^{S,T} - N_j^{S,T}\| \leq \rho, \ \ \forall j \\ outlier & \text{otherwise.} \end{cases} \end{aligned} \tag{5}$$

where $j = 1, \ldots, X$, and $X = |D| = |\widehat{D}|$.

Our proposed pose ensemble aims to fuse two predictions (N^S, N^T) to estimate the optimal pose. Although this fusion could be conducted at the input level before obtaining the T from θ_p, we found that our proposal of fusing at the output level after getting T produces slightly better results (see the ablation study, Sec. 4.5). We first estimate each pose (T^S, T^T) from each prediction (N^S, N^T) using θ_p. We then calculate the number of inliers $(inliers^S, inliers^T)$ from each pose using Eq. (5) to evaluate the confidence of the pose. We assume that the higher this confidence (*i.e.*, the more inliers) is, the more accurate the pose estimation will be. Therefore:

$$T_{out} = \begin{cases} T^S & \text{if } inliers^S > inliers^T, \\ T^T & \text{otherwise.} \end{cases} \tag{6}$$

We found that this simple pose-aware ensemble method is more effective for generating high-quality pseudo labels and inference than other ensemble methods (Sec. 4.5).

Another crucial part of our method is a self-training loss. The total loss that we use to update the student network combines this self-training L_d loss and the pseudo label loss L_{pl}:

$$L_{tta} = \lambda_d L_d + \lambda_{pl} L_{pl}. \tag{7}$$

The first term is self-training loss L_d, which helps learn the distribution of the target domain and reduce the domain gap through T_{out}, which is obtained from the pose ensemble and observed D. This loss is given by:

$$L_d = L_{CE}(N_e^S, \text{Enc}(\widehat{D}_e)), \tag{8}$$

where N_e^S is the inlier-only student NOCS map, $\widehat{D}_e$ is the resulting inliner-only transformed depth point cloud in NOCS space, and $\text{Enc}(\cdot)$ is the one-hot encoding to enable cross-entropy loss.

The second term

$$L_{pl} = L_{CE}(N_e^S, N_e^T) \tag{9}$$

is the pseudo label loss from UDA-COPE [15], where N_e^T is the inlier-only teacher NOCS map. Since $N^S = \theta^S(x_t)$, and $\hat{y} = \theta^T(x_t)$, L_{pl} in Eq. (9) is the same as the loss in Eq. (3), except that here only inliers are considered.

4. Experiments

4.1. Dataset

We utilize two widely used category-level pose estimation datasets as the source and target domain, respectively. The *source data* is the Context-Aware MixEd ReAlity (CAMERA) dataset [43], generated by rendering and compositing synthetic objects into real scenes while considering the context. CAMERA consists of 275K RGB-D images as a training set with 1,085 object instances chosen from six categories: bottle, bowl, camera, can, laptop and mug. We use the REAL dataset [43] as the *target domain*. The target data consists of 4,300 real-world images of seven scenes for training and 2,750 real-world images of six scenes for evaluation. We refer to REAL evaluation set as REAL275. Our TTA methods do not use the target training set (4,300 images with seven scenes) and only use the evaluation set (2,750 images with six scenes, REAL275) for test-time adaptation, *e.g., TENT [42]* process evaluation set in a sequential, online manner. On the other hand, unsupervised methods such as [5, 16, 29] use the target training set for updating the models without any restrictions on how much and how long they use the information.

4.2. Implementation Details

Object Segmentation. We use Mask R-CNN [8] to obtain the object area in 2D image. We use the identical results of Mask R-CNN for a fair comparison with previous methods [3, 17, 36] for semi-supervised and unsupervised settings. For the semi-supervised setting, Mask R-CNN has trained on the source, and target domain supervised manner. For the unsupervised setting, Mask R-CNN is trained on only the source domain. The detected area of the image resizes to 192 x 192 image resolution as the teacher or student model input.

Table 2. **Quantitative comparisons with state-of-the-art methods on the REAL275 dataset.**

Method	Supervised		Unsupervised		mAP (↑)					
	Source	Target	Target	Online	IoU_{50}	IoU_{75}	5° 2cm	5° 5cm	10° 2cm	10° 5 cm
Source (Supervised)										
Metric Scale [14]	2D/3D				54.6	8.4	2.2	5.4	10.1	25.0
SPD [36]	2D/3D				50.5	17.0	11.5	12.1	33.0	37.9
Source (Supervised) & Target (Supervised)										
NOCS [43]	2D/3D	2D/3D			47.2	9.4	7.2	10.0	13.8	25.2
SPD [36]	2D/3D	2D/3D			68.5	27.0	19.5	21.6	43.5	54.0
DualPoseNet [17]	2D/3D	2D/3D			—	30.8	29.3	35.9	50.0	66.8
CR-Net [44]	2D/3D	2D/3D			—	33.2	27.8	34.3	47.2	60.8
SGPA [3]	2D/3D	2D/3D			—	37.1	35.9	39.6	61.3	70.7
Source (Supervised) & Target (Semi-Supervised)										
SSC-6D [29]	2D/3D	2D	3D	✗	73.0	—	16.8	19.6	44.1	54.5
RePoNet [5]	2D/3D	2D	3D	✗	76.0	—	30.7	33.9	—	63.0
UDA-COPE [15]	2D/3D	2D	3D	✗	75.5	34.4	30.5	34.9	57.0	66.1
Self-DPDN [16]	2D/3D	2D	3D	✗	75.2	41.6	**39.5**	**45.0**	63.3	72.2
TTA-COPE (Ours)	2D/3D	2D	3D	✓	**78.7**	**43.5**	33.3	38.1	**64.3**	**75.1**
Source (Supervised) & Target (Unsupervised)										
Self-DPDN [16]	2D/3D		2D/3D	✗	67.2	**43.9**	**39.0**	**46.7**	**61.8**	**73.4**
TTA-COPE (Ours)	2D/3D		2D/3D	✓	**69.1**	39.7	30.2	35.9	61.7	73.2

Student and Teacher. The student and teacher models have identical design in 2D and 3D branches from UDA-COPE [15]. We utilize the PSPNet [53] with ResNet34 [9] backbone for the 2D image feature extraction. For a 3D branch, we use the MinkowskiNet [4] and utilize sparse convolution operation with a 5cm voxel size. Our NOCS representation uses the classification with 32 bins [43] instead of direct regression. During the pretraining stage, we train our model on the source data for 50 epochs using the Adam optimizer by initializing the learning rate of 1e-4 with a batch size of 32. The learning rate was reduced by a ratio of 0.6 (at 15k iterations), 0.3 (at 30k iterations), 0.1 (at 45k iterations), and 0.01 (at 60k iterations). During test-time adaptation, our student model uses the same learning rate, and the teacher smoothly updates using momentum update with $\gamma = 0.99$. Given target data, we first update the model every iteration and then estimate the pose. We set $\lambda_{CE} = 1.0$, $\lambda_{C} =$ 1e-6, $\lambda_{d} = 1.0$, $\lambda_{pl} = 1.0$ for our experiments. The point filtering threshold ρ was set to 0.05 for all experiments.

Metrics. To evaluate the performance of 3D object detection and 6D pose estimation, we follow the previous pose and size evaluation metric from Wang *et al*. [43]. We report the mean average precision (mAP) at the 50% and 75% intersection over union (IoU) thresholds for 3D object detection. We also report mAP for 6D object pose evaluation w.r.t. rotation and translation errors, where e.g. the 5° 5cm metric describes the percentage of pose predictions where the error is less than both 5° and 5cm and the same for other thresholds. We recalculated all the metrics with the improved code.

Table 3. **Quantitative comparisons with TTA baselines for category-level object pose estimation on the REAL275 dataset.**

Method	Unsupervised		mAP (↑)		
	Target 3D	Target 2D	IoU_{75}	5° 2cm	5° 5 cm
TENT [42] - Eq. (2)	✓		33.7	25.9	29.1
PL [35] - Eqs. (3)–(4)	✓		39.9	29.7	34.9
PL-F [15] - Eqs. (3)–(5)	✓		41.1	31.4	36.3
TTA-COPE (Ours)	✓		**43.5**	**33.3**	**38.1**
TENT [42] - Eq. (2)	✓	✓	32.3	26.8	31.2
PL [35] - Eqs. (3)–(4)	✓	✓	36.0	26.9	33.1
PL-F [15] - Eqs. (3)–(5)	✓	✓	36.5	28.0	34.0
TTA-COPE (Ours)	✓	✓	**39.7**	**30.2**	**35.9**

4.3. Comparison with state-of-the-art

Table 2 summarizes quantitative results on the REAL275 dataset under different settings: 1) supervised only on source data, 2) supervised on source and target data, 3) supervised on source data and semi-supervised on target data, and 4) supervised on source data and unsupervised on target data. Not surprisingly, methods following 2) such as SPD [36] perform better than setting 1) since they have the benefit of accessing both source and target data. Recent unsupervised domain adaptation (UDA) methods [15, 16] show remarkable results compared to state-of-the-art supervised methods without using target 3D labels (semi-supervised setting). Our method shows state-of-the-art results in IoU metrics under the semi-supervised setting and even outperforms the recent supervised methods SGPA [3] by a large margin (6.4 mAP in IoU_{75}). Our TTA-COPE uses less time and data to train the target domain because of the advantage of test-time adatpation but achieves comparable

Table 4. **Ablation study on variants of pose ensemble methods and self-training loss under the semi-supervised setting.**

Method	Pose Ensemble					mAP (↑)					
	Input	Output	Inference	Pseudo Label	L_d (8)	IoU_{50}	IoU_{75}	5° 2cm	5° 5 cm	10° 2cm	10° 5 cm
LB						76.2	37.5	29.1	34.6	62.1	73.2
(1)	Argmax Match		✓			71.1	27.5	29.6	34.3	60.7	71.7
(2)	Softmax Avg.		✓			**78.7**	40.4	32.1	37.3	64.0	74.7
(3)	Softmax Max		✓			77.6	40.9	31.6	36.4	63.4	74.3
(4)		Softmax Max	✓			77.5	41.1	32.1	37.1	64.3	74.6
(5)		Inliers Max	✓			77.6	41.2	32.8	37.6	64.4	74.9
(6)		Inliers Max	✓		✓	78.2	42.9	32.4	37.8	64.1	74.7
(7)	Argmax Match			✓		78.3	40.7	31.7	36.4	63.2	73.9
(8)	Softmax Avg.			✓		78.6	41.1	31.5	36.2	63.6	74.5
(9)	Softmax Max			✓		78.1	41.2	31.9	36.5	63.5	74.4
(10)		Softmax Max		✓		78.6	41.2	31.8	36.3	63.5	74.4
(11)		Inliers Max		✓		78.2	41.3	31.6	36.2	63.6	74.3
(12)		Inliers Max		✓	✓	**78.7**	43.0	32.6	37.0	63.5	73.9
Ours		Inliers Max	✓	✓	✓	**78.7**	**43.5**	**33.3**	**38.1**	**64.6**	**75.1**

results to the SOTA method, Self-DPDN [16]. Ours takes about 31 minutes (Table 5) for TTA and is 58x faster than the Self-DPDN, which takes about 30 hours for training in the target domain, excluding inference time.

4.4. Comparison with TTA baselines

Table 3 summarizes results for the different design test-time adaptation (TTA) baselines in semi-supervised and unsupervised settings. TENT [42] has been designed for general unsupervised settings, but it is not specially designed for the object pose estimation task and naturally observes overall poor performance compared to other TTA baselines. Mean teacher with pseudo labels (PL) [35] performs better than TENT [42] in semi-supervised and unsupervised settings, and we believe that training the model using pseudo labels with momentum update (Eqs. (3)–(4)) provides more stable training signals by reducing the error accumulation than entropy minimization (2). However, using pseudo labels without filtering (PL) provides unreliable labels as the ground truth, we found that pseudo label filtering (PL-F) [15] performs better than PL, which indicates that noise filtering (Eq. (5)) improves the results for the student as expected. Finally, our TTA-COPE achieves state-of-the-art (SOTA) performance among all TTA baselines under both semi-supervised and unsupervised settings.

4.5. Ablation Study

In this section, we conduct experiments to evaluate the efficacy of the pose ensemble and self-training loss under a semi-supervised setting.

Input/Output-Level Pose Ensemble. As mentioned in the Sec. 3.3, we compare the different pose ensemble methods given predictions of teacher and student models (N^T, N^S) and answer the following question: Which ensemble is the most effective for test-time adaptation? We categorize two ensemble techniques:

1) Input-level ensemble that fuses two NOCS map predictions (N^T, N^S) and makes an accurate single NOCS map N to estimate poses using the Umeyama algorithm θ_p. For the input-level ensemble that fuses two predictions (N^T, N^S) into one prediction N, we compare three strategies: Argmax Match, Softmax Average, and Softmax Max operations [12, 33].

2) We ensemble output-level predictions (N^T, N^S) to estimate each pose (T^T, T^S) and choose the best pose predictions under specific criteria. Unlike the input ensemble, only the Softmax Max operation is valid for the output-level ensemble due to the nonlinearity of poses.

Table 4 summarizes our ensemble ablation study results. Table 4-(1-3) shows that the Softmax Avg. and Softmax Max perform better than the Argmax Match operation among input-level pose ensembles. If we compare the same operation (Softmax Max) in the input-level, Table 4-(3), and output-level, Table 4-(4), the output-level ensemble shows slightly better performance since the output-level operation is directly related to the pose results. Our proposed pose ensemble, Table 4-(5), considers pose-aware confidence by choosing a higher number of inlier points and achieves the best performance in all pose ensembles in Table 4-(1-5).

Use of Pose Ensemble. Our pose ensemble can be used for inference Table 4-(1-6, Ours) as well as generating pseudo labels Table 4-(7-12, Ours). The results show that our pose ensemble is helpful for both cases. In particular, applying our ensemble to inferences Table 4-(5) has been shown to be more effective than application to generating pseudo labels Table 4-(11) in pose metric. We also observe that our pose ensemble used for inference or pseudo labels improves overall metrics compared to the lower bound (LB) that is only trained on source data without test-time adaptation.

Effect of self-training loss. The results of Table 4-(6) and Table 4-(12) show the effect of self-training loss Eq. (8). It yields an improvement of more than 1.7 mAP compared

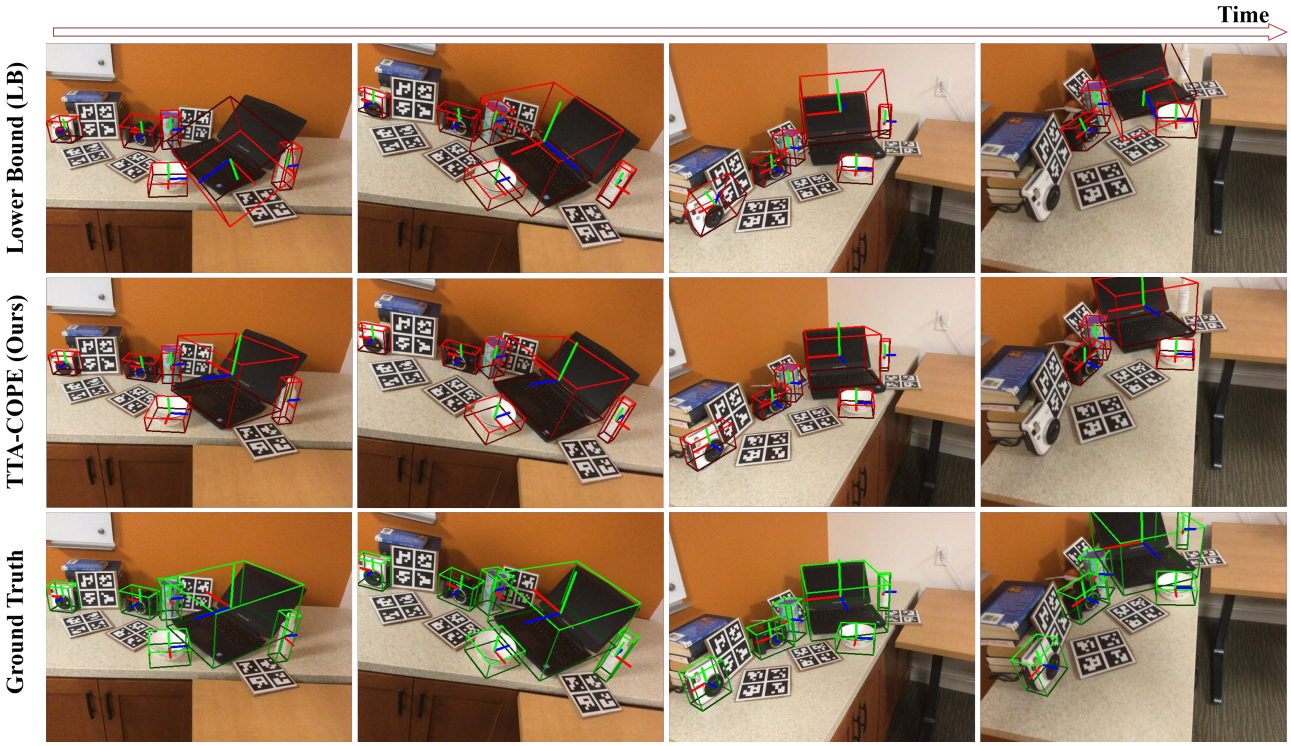

Figure 3. **Qualitative comparison with the lower bound (LB) and our TTA-COPE under the semi-supervised setting.**

Table 5. **Ablation study on different updating intervals.**

Method	TTA Time	IoU_{75}	5° 2cm	5° 5 cm
Interval=1 (Ours)	31 min	43.5	33.3	38.1
Interval=10	16 min	41.8	32.7	37.7
Interval=20	15 min	41.1	31.9	37.1

to without self-training loss results of Table 4-(5, 11) in the IoU_{75} metric. This indicates that the self-learning loss helps to learn the distribution of the target domain and reduces the domain shift by comparing the predicted NOCS map against the observed point cloud in the target domain. The result in the last column is our proposed model for test-time adaptation, Fig. 3 shows qualitative results against the lower bound method.

Updating Interval. TTA is inevitably slower than simple inference since additional training is required. However, reducing the update interval enables faster TTA time than TTA baselines because of reduced training time. Table 5 shows the difference in TTA speed and performance according to the updating interval. Specifically, the interval of every 10 frames improves the speed of TTA roughly two-fold compared to Interval 1 (Ours), with a marginal performance drop. We also increase intervals to 20, but it does not show as much improvement as before since most of the bottleneck arises from inference time.

5. Conclusion

We have proposed TTA-COPE, a test-time adaptation method for category-level object pose estimation that addresses the source-to-target domain shift without accessing source data at test time and without labeled target data. Specifically, we designed a pose ensemble method with self-training for test-time adaptation that simultaneously uses the teacher-student model to generate robust pseudo labels and estimate accurate poses for inference. We explore limitations of several test-time adaptation baselines and show that the proposed method achieves state-of-the-art performance. We demonstrate the benefits of our proposed pose ensemble and self-training loss with extensive studies in both semi-supervised and unsupervised settings.

To the best of our knowledge, TTA-COPE is the first approach that tries to solve test-time adaptation for category-level object pose estimation. Since our method currently focuses on generating 6D pose labels, it does not affect 2D labels and the segmentation model. In future work, when jointly considering 2D label and Mask R-CNN, greater performance improvement and stable test-time adaptation would be possible. Also, our pose estimation relies on non-differentiable pose estimation, and as such we could benefit from a differentiable pose estimation method [30].

Acknowledgment

This work was part of Taeyeop Lee's internship at NVIDIA and was also partially supported by an Institute of Information & communications Technology Planning & Evaluation(IITP) grant funded by the Korean government(MSIT) (No.2020-0-00440).

References

[1] Adel Ahmadyan, Liangkai Zhang, Artsiom Ablavatski, Jianing Wei, and Matthias Grundmann. Objectron: A large scale dataset of object-centric videos in the wild with pose annotations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7822–7831, 2021. 1

[2] Dengsheng Chen, Jun Li, Zheng Wang, and Kai Xu. Learning canonical shape space for category-level 6D object pose and size estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11973–11982, 2020. 1, 2, 3

[3] Kai Chen and Qi Dou. SGPA: Structure-guided prior adaptation for category-level 6D object pose estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2773–2782, 2021. 1, 2, 3, 5, 6

[4] Christopher Choy, JunYoung Gwak, and Silvio Savarese. 4d spatio-temporal convnets: Minkowski convolutional neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3075–3084, 2019. 6

[5] Yang Fu and Xiaolong Wang. Category-level 6d object pose estimation in the wild: A semi-supervised learning approach and a new dataset. *arXiv preprint arXiv:2206.15436*, 2022. 2, 3, 5, 6

[6] Wei Gao and Russ Tedrake. kpam 2.0: Feedback control for category-level robotic manipulation. *IEEE Robotics and Automation Letters (RA-L)*, 6(2):2962–2969, 2021. 1

[7] Richard Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2003. 2

[8] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2961–2969, 2017. 3, 5

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016. 6

[10] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning (ICML)*, pages 448–456, 2015. 3

[11] Muhammad Zubair Irshad, Sergey Zakharov, Rares Ambrus, Thomas Kollar, Zsolt Kira, and Adrien Gaidon. ShAPO: Implicit representations for multi-object shape, appearance, and pose optimization. *arXiv preprint arXiv:2207.13691*, 2022. 3

[12] Maximilian Jaritz, Tuan-Hung Vu, Raoul de Charette, Emilie Wirbel, and Patrick Pérez. xmuda: Cross-modal unsupervised domain adaptation for 3d semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12605–12614, 2020. 7

[13] Yann Labbé, Lucas Manuelli, Arsalan Mousavian, Stephen Tyree, Stan Birchfield, Jonathan Tremblay, Justin Carpentier, Mathieu Aubry, Dieter Fox, and Josef Sivic. MegaPose: 6d pose estimation of novel objects via render & compare. In *6th Annual Conference on Robot Learning (CoRL)*, 2022. 1

[14] Taeyeop Lee, Byeong-Uk Lee, Myungchul Kim, and In So Kweon. Category-level metric scale object shape and pose estimation. *IEEE Robotics and Automation Letters (RA-L)*, 6(4):8575–8582, 2021. 2, 6

[15] Taeyeop Lee, Byeong-Uk Lee, Inkyu Shin, Jaesung Choe, Ukcheol Shin, In So Kweon, and Kuk-Jin Yoon. UDA-COPE: Unsupervised domain adaptation for category-level object pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14891–14900, 2022. 2, 3, 4, 5, 6, 7

[16] Jiehong Lin, Zewei Wei, Changxing Ding, and Kui Jia. Category-level 6d object pose and size estimation using self-supervised deep prior deformation networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 19–34, 2022. 2, 3, 4, 5, 6, 7

[17] Jiehong Lin, Zewei Wei, Zhihao Li, Songcen Xu, Kui Jia, and Yuanqing Li. DualPoseNet: Category-level 6D object pose and size estimation using dual pose network with refined learning of pose consistency. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3560–3569, 2021. 1, 2, 3, 5, 6

[18] Yunzhi Lin, Jonathan Tremblay, Stephen Tyree, Patricio A. Vela, and Stan Birchfield. Single-stage keypoint-based category-level object pose estimation from an rgb image. In *IEEE International Conference on Robotics and Automation (ICRA)*. ICRA, 2022. 1

[19] Shichen Liu, Tianye Li, Weikai Chen, and Hao Li. Soft rasterizer: A differentiable renderer for image-based 3d reasoning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 7708–7717, 2019. 3

[20] Shaohui Liu, Yinda Zhang, Songyou Peng, Boxin Shi, Marc Pollefeys, and Zhaopeng Cui. DIST: Rendering deep implicit signed distance function with differentiable sphere tracing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2019–2028, 2020. 3

[21] Yuang Liu, Wei Zhang, and Jun Wang. Source-free domain adaptation for semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1215–1224, 2021. 2, 3

[22] Massimiliano Mancini, Hakan Karaoguz, Elisa Ricci, Patric Jensfelt, and Barbara Caputo. Kitting in the wild through online domain adaptation. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1103–1109. IEEE, 2018. 3

[23] Eric Marchand, Hideaki Uchiyama, and Fabien Spindler. Pose estimation for augmented reality: A hands-on survey. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 22(12):2633–2651, 2015. 1

[24] Eitan Marder-Eppstein. Project Tango. In *ACM SIGGRAPH Real-Time Live!*, page 25, 2016. 1, 3

[25] Arsalan Mousavian, Clemens Eppner, and Dieter Fox. 6-DoF GraspNet: Variational grasp generation for object manipulation. In *Proceedings of the IEEE/CVF International*

Conference on Computer Vision (ICCV), pages 2901–2910, 2019. 1, 3

[26] Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Yaofo Chen, Shijian Zheng, Peilin Zhao, and Mingkui Tan. Efficient test-time model adaptation without forgetting. In *International Conference on Machine Learning (ICML)*, pages 16888–16905, 2022. 3

[27] Kiru Park, Timothy Patten, and Markus Vincze. Pix2Pose: Pixel-wise coordinate regression of objects for 6D pose estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 7668–7677, 2019. 1

[28] 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 (CVPR)*, pages 4561–4570, 2019. 1

[29] Wanli Peng, Jianhang Yan, Hongtao Wen, and Yi Sun. Self-supervised category-level 6D object pose estimation with deep implicit shape representation. In *Proceedings of the Association for the Advancement of Artificial Intelligence (AAAI)*, volume 36, pages 2082–2090, 2022. 2, 3, 5, 6

[30] Luis Pineda, Taosha Fan, Maurizio Monge, Shobha Venkataraman, Paloma Sodhi, Ricky T. Q. Chen, Joseph Ortiz, Daniel DeTone, Austin Wang, Stuart Anderson, Jing Dong, Brandon Amos, and Mustafa Mukadam. Theseus: A library for differentiable nonlinear optimization. *Advances in Neural Information Processing Systems*, 2022. 8

[31] Mihir Prabhudesai, Sujoy Paul, Sjoerd van Steenkiste, Mehdi SM Sajjadi, Anirudh Goyal, Deepak Pathak, Katerina Fragkiadaki, Gaurav Aggarwal, and Thomas Kipf. Test-time adaptation with slot-centric models. 2022. 3

[32] Martin Runz, Maud Buffier, and Lourdes Agapito. MaskFusion: Real-time recognition, tracking and reconstruction of multiple moving objects. In *IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 10–20, 2018. 1

[33] Inkyu Shin, Yi-Hsuan Tsai, Bingbing Zhuang, Samuel Schulter, Buyu Liu, Sparsh Garg, In So Kweon, and Kuk-Jin Yoon. MM-TTA: Multi-modal test-time adaptation for 3D semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16928–16937, 2022. 3, 7

[34] Junha Song, Jungsoo Lee, In So Kweon, and Sungha Choi. Ecotta: Memory-efficient continual test-time adaptation via self-distilled regularization. *arXiv preprint arXiv:2303.01904*, 2023. 3

[35] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *Advances in Neural Information Processing Systems (NeurIPS)*, 30, 2017. 2, 4, 6, 7

[36] Meng Tian, Marcelo H Ang Jr, and Gim Hee Lee. Shape prior deformation for categorical 6D object pose and size estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 1, 2, 3, 5, 6

[37] Jonathan Tremblay, Aayush Prakash, David Acuna, Mark Brophy, Varun Jampani, Cem Anil, Thang To, Eric Cameracci, Shaad Boochoon, and Stan Birchfield. Training deep networks with synthetic data: Bridging the reality gap by domain randomization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 969–977, 2018. 1

[38] Jonathan Tremblay, Thang To, Balakumar Sundaralingam, Yu Xiang, Dieter Fox, and Stan Birchfield. Deep object pose estimation for semantic robotic grasping of household objects. In *Conference on Robot Learning (CoRL)*, 2018. 1

[39] Stephen Tyree, Jonathan Tremblay, Thang To, Jia Cheng, Terry Mosier, Jeffrey Smith, and Stan Birchfield. 6-DoF pose estimation of household objects for robotic manipulation: An accessible dataset and benchmark. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2022. 1

[40] Shinji Umeyama. Least-squares estimation of transformation parameters between two point patterns. *IEEE Transactions on Pattern Analysis & Machine Intelligence (TPAMI)*, 13(04):376–380, 1991. 2

[41] 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 (CVPR)*, pages 3343–3352, 2019. 1

[42] Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. In *International Conference on Learning Representations (ICLR)*, 2021. 2, 3, 4, 5, 6, 7

[43] He Wang, Srinath Sridhar, Jingwei Huang, Julien Valentin, Shuran Song, and Leonidas J Guibas. Normalized object coordinate space for category-level 6D object pose and size estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2642–2651, 2019. 1, 2, 3, 5, 6

[44] Jiaze Wang, Kai Chen, and Qi Dou. Category-level 6D object pose estimation via cascaded relation and recurrent reconstruction networks. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021. 1, 2, 3, 6

[45] Pengyuan Wang, HyunJun Jung, Yitong Li, Siyuan Shen, Rahul Parthasarathy Srikanth, Lorenzo Garattoni, Sven Meier, Nassir Navab, and Benjamin Busam. Phocal: A multi-modal dataset for category-level object pose estimation with photometrically challenging objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 21222–21231, 2022. 1

[46] Qin Wang, Olga Fink, Luc Van Gool, and Dengxin Dai. Continual test-time domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7201–7211, 2022. 2, 3

[47] Bowen Wen and Kostas Bekris. BundleTrack: 6D pose tracking for novel objects without instance or category-level 3D models. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 8067–8074, 2021. 1

[48] Bowen Wen, Wenzhao Lian, Kostas Bekris, and Stefan Schaal. You only demonstrate once: Category-level manipulation from single visual demonstration. *Robotics: Science and Systems (RSS)*, 2022. 1

[49] Bowen Wen, Chaitanya Mitash, Baozhang Ren, and Kostas E Bekris. se (3)-tracknet: Data-driven 6d pose tracking by calibrating image residuals in synthetic domains. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 10367–10373. IEEE, 2020. 1

[50] Bowen Wen, Chaitanya Mitash, Sruthi Soorian, Andrew Kimmel, Avishai Sintov, and Kostas E Bekris. Robust, occlusion-aware pose estimation for objects grasped by adaptive hands. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 6210–6217. IEEE, 2020. 1

[51] Yu Xiang, Tanner Schmidt, Venkatraman Narayanan, and Dieter Fox. PoseCNN: A convolutional neural network for 6D object pose estimation in cluttered scenes. *Robotics: Science and Systems (RSS)*, 2018. 1

[52] He Yisheng, Wang Yao, Fan Haoqiang, Chen Qifeng, and Sun Jian. Fs6d: Few-shot 6d pose estimation of novel objects. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 1

[53] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2881–2890, 2017. 6