Manifold-Aware Self-Training for Unsupervised Domain Adaptation on Regressing 6D Object Pose

Yichen Zhang[1] , **Jiehong Lin**[1] , **Ke Chen**[1,2,*] , **Zelin Xu**[1] , **Yaowei Wang**[2] and **Kui Jia**[1,2,*]

[1]South China University of Technology
[2]Peng Cheng Laboratory

{eezyc, lin.jiehong, eexuzelin}@mail.scut.edu.cn, {chenk, kuijia}@scut.edu.cn, wangyw@pcl.ac.cn

Abstract

Domain gap between synthetic and real data in visual regression (*e.g.* 6D pose estimation) is bridged in this paper via global feature alignment and local refinement on the coarse classification of discretized anchor classes in target space, which imposes a piece-wise target manifold regularization into domain-invariant representation learning. Specifically, our method incorporates an explicit self-supervised manifold regularization, revealing consistent cumulative target dependency across domains, to a self-training scheme (*e.g.* the popular Self-Paced Self-Training) to encourage more discriminative transferable representations of regression tasks. Moreover, learning unified implicit neural functions to estimate relative direction and distance of targets to their nearest class bins aims to refine target classification predictions, which can gain robust performance against inconsistent feature scaling sensitive to UDA regressors. Experiment results on three public benchmarks of the challenging 6D pose estimation task can verify the effectiveness of our method, consistently achieving superior performance to the state-of-the-art for UDA on 6D pose estimation. Code is available at https://github.com/Gorilla-Lab-SCUT/MAST.

1 Introduction

The problems of visual regression such as estimation of 6D pose of object instances (*i.e.* their orientation and translation with respect to the camera optical center) and configuration of human body parts given RGB images are widely encountered in numerous fields such as robotics [Collet *et al.*, 2011; He *et al.*, 2020], augmented reality [Marchand *et al.*, 2015] and autonomous driving [Geiger *et al.*, 2012; Chen *et al.*, 2017], which can be typically addressed by learning a single or multi-output regression mapping on deep representations of visual observations. Recent regression algorithms have gained remarkable success to handle with inconsistent lighting conditions and heavy occlusions in-between foreground
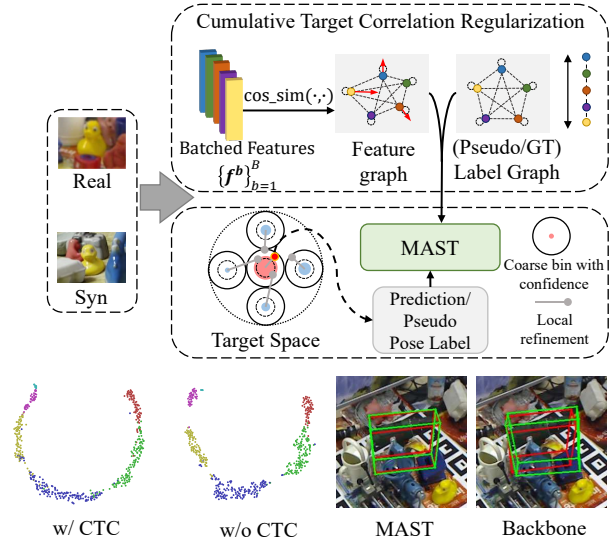

Figure 1: Illustration of the proposed Manifold-Aware Self Training (MAST) for UDA on 6D pose estimation. Top: a novel cumulative target correlation (CTC) regularization on representation learning. Middle: one bin with the highest confidence (*i.e.* highlighted with the largest red dish) with further local refinement (*i.e.* the gray lines) are adopted to generate pseudo pose labels in our MAST. Bottom: the t-SNE visualization of feature distribution w/ and w/o the proposed CTC, which can verify the effectiveness of introduction of target manifolds with a more smooth distribution; and comparison with our MAST and its backbone with an example from the Occluded LineMOD dataset is given.

and contextual objects in uncontrolled and cluttered environment, owing to recent development of representation learning of visual regression, such as introduction of self-supervised regularization [Lin *et al.*, 2022] and powerful network architectures [He *et al.*, 2021].

In those regression tasks, visual observations, *i.e.* RGB images, can be easily acquired in practice or directly collected from the Internet, but it is laborious or even unfeasible for manual noise-free annotation with continuous targets. As a result, the size of real training data with precise labels is typically limited and less scalable, *e.g. eggbox* and *holepuncher* training samples in the LineMOD [Hinterstoisser *et al.*, 2012] for 6D pose estimation, which increases

---

*Corresponding authors.

the difficulty of learning good representations. The synthesis of images can be a powerful solution to cope with data sparsity, which can be gained via photorealistic rendering [Hodaň *et al.*, 2019] with CAD models. However, domain discrepancy between synthetic and real data, *e.g.* appearance difference between CAD models and real objects, scene illumination, and systematic imaging noises, can lead to collapse of regression performance, which encourages the practical setting of unsupervised domain adaptation on visual regression (UDAVR), *i.e.* samples in the source and target domains cannot satisfy the i.i.d. condition.

Different from the widely-investigated problem of unsupervised domain adaptation on visual classification (UDAVC) [Gopalan *et al.*, 2011; Zou *et al.*, 2018; Zou *et al.*, 2021], only a few works [Chen *et al.*, 2021; Lee *et al.*, 2022] have explored the vital factors of representation learning of visual regression that different from classification in the context of UDA. [Chen *et al.*, 2021] revealed and exploited the sensitivity of feature scaling on domain adaptation regression performance to regularize representation learning, which can achieve promising results to bridge domain gap. We argue that *cumulative dependent nature* and *piece-wise manifolds* in target space are two key factors of UDA regression yet missing in the existing algorithms. To this end, this paper proposes a **M**anifold-**A**ware **S**elf-**T**raining (MAST) scheme to decompose the problem of learning a domain-invariant regression mapping into a combination of a feature-scaling-robust globally coarse classification of discretized target anchors via self-training based feature alignment and a locally regression-based refinement less sensitive to inconsistent feature scale, as shown in Figure 1.

For exploiting the cumulative dependent nature of regression targets different from those in classification, the self-training method (*e.g.* the self-paced self-training [Zou *et al.*, 2018]) originally designed for the UDAVC problem is now adapted to the coarse classification on discretization of continuous target space, with incorporating a novel piece-wise manifold regularization on domain-invariant representation learning, namely a self-supervised cumulative target correlation regularization. Intuitively, appearance ambiguities across domains in representation learning can be mitigated via leveraging consistent target correlation under certain distance metrics in target space (*e.g.* the Euclidean distance in $R(3)$ translation space). Furthermore, considering the risk of sensitivity to varying feature scaling in the UDAVR problem [Chen *et al.*, 2021], learning unified local regression functions with those shared features of the classification of discretized target bins (typically having inconsistent feature scales) can achieve superior robustness against large scale variations of transferable representations. Extensive experiments on three popular benchmarks of the challenging UDA on 6D pose estimation can confirm the effectiveness of our MAST scheme, consistently outperforming the state-of-the-art.

The novelties of our paper are summarized as follows.

- This paper proposes a novel and generic manifold-aware self-training scheme for unsupervised domain adaptation on visual regression, which exploits cumulative correlation and piece-wise manifolds in regression target space for domain-invariant representation learning.

- Technically, a novel cumulative target correlation regularization is proposed to regularize the self-training algorithm on coarse classification with latent dependency across regression targets, while local refinement can be achieved via learning implicit functions to estimate residual distance to the nearest anchor within local target manifolds in a unified algorithm.

- Experiment results on multiple public benchmarks of UDA on 6D pose estimation can verify consistent superior performance of our scheme to the state-of-the-art UDA pose regressors.

2 Related Works

6D Pose Estimation. The problem of estimating 6D pose of object instances within an RGB image (optionally with a complementary depth image) is active yet challenging in robotics and computer vision. With the rise of deep learning, recent methods for predicting 6D poses can be divided into two main groups – keypoint-based [Peng *et al.*, 2019; Zakharov *et al.*, 2019] and regression based [Xiang *et al.*, 2018; Labbé *et al.*, 2020; Wang *et al.*, 2021b]. The former relied on learning a 2D-to-3D correspondence mapping between object keypoints in 3D space and their 2D projection on images with the Perspective-n-Point (PnP) [Fischler and Bolles, 1981]. Such a correspondence can be achieved by either detecting a limited size of landmarks [Tekin *et al.*, 2018; Peng *et al.*, 2019] or pixel-wise voting from a heatmap [Park *et al.*, 2019; Zakharov *et al.*, 2019]. The latter concerned on deep representation learning for direct pose regression with the point-matching loss for optimizing output pose [Xiang *et al.*, 2018; Labbé *et al.*, 2020] or proposing a differentiable PnP paradigm in an end-to-end training style [Wang *et al.*, 2021b; Chen *et al.*, 2022a]. Alternatively, the problem can also be formulated into ordinal classification via discretization of $SE(3)$ space into class bins [Su *et al.*, 2015; Kehl *et al.*, 2017]. To alleviate representation ambiguities, the estimated 6D pose of objects can be further refined via either an iterative refinement with residual learning [Li *et al.*, 2018b; Manhardt *et al.*, 2018] or simply the Iterative Closest Point [Xiang *et al.*, 2018], while some work introduced cross-view fusion based refinement [Labbé *et al.*, 2020; Li *et al.*, 2018a]. Existing refinement strategies are typically employed as a post-processing step following the main module of 6D pose estimation, some of which such as [Li *et al.*, 2018b; Xu *et al.*, 2022] can be designed in an end-to-end learning cascade to obtain significant performance gain, but they are not designed for bridging domain gap and therefore cannot ensure good performance under the UDAVR setting. Alternatively, [Li *et al.*, 2018a] introduced a combined scheme of both coarse classification and local regression-based refinement simultaneously, which is similar to our MAST method. However, the main differences lie in the introduction of the cumulative target correlation regularization in our scheme to encourage domain-invariant pose representations revealing the dependent nature of regression targets.

Unsupervised Domain Adaptation on Visual Regression. Most of regression methods [Xu *et al.*, 2019; Bao *et al.*, 2022] employ annotated real data for model training, but

manual annotations on real data are usually laboriously expensive or even unfeasible. Lack of sufficient annotated real data encourages the practical setting of Simulation-to-Reality (Sim2Real) UDAVR, *i.e.* learning a domain-agnostic representation given annotated synthetic data as source domain and unlabeled real data as target domain. A simple yet effective way to narrow Sim2Real domain gap can rely on domain randomization [Kehl *et al.*, 2017; Manhardt *et al.*, 2018], while recent success of self-supervised learning for UDAVC [Zou *et al.*, 2021; Yue *et al.*, 2021] inspired a number of self-supervised regressors [Wang *et al.*, 2021a; Yang *et al.*, 2021] in the context of Regression. Self6D [Wang *et al.*, 2020] and its extension Self6D++ [Wang *et al.*, 2021a] leveraged a differentiable renderer to conduct self-supervised visual and geometrical alignment on visible and amodal object mask predictions. Bao *et al.* [Bao *et al.*, 2022] introduced a self-supervised representation learning of relative rotation estimation to adapt one gaze regressor to the target domain. Zhang *et al.* [Zhang *et al.*, 2021] utilized a Graph Convolutional Network to model domain-invariant geometry structure among key-points, which is applied to guide training of the object pose estimator on real images. These mentioned algorithms were designed for only one specific task and cannot be directly applied to other visual regression problems. [Chen *et al.*, 2021] proposed the representation subspace distance (RSD) generic to multiple UDAVR problems, but cannot perform well on the challenging task having severe representation ambiguities, *e.g.* 6D pose estimation investigated in this paper (see Table 3). In contrast, the proposed MAST scheme is generic to UDAVR owing to exploiting explicit target correlation in the style of local manifolds to regularize deep representation learning agnostic to domains.

Self-Training. Self-training methods utilize a trained model on labeled data to make predictions of unannotated data as pseudo labels [Lee and others, 2013] (*i.e.* supervision signals assigned to unlabeled data), which is widely used in semi-supervised learning [Lee and others, 2013; Sohn *et al.*, 2020] and UDA [Roy *et al.*, 2019]. [Sohn *et al.*, 2020] generated pseudo labels from weakly augmented images, which are adopted as supervision of strongly augmented variants in semi-supervised learning; similar scripts are shared with the noisy student training [Xie *et al.*, 2020]. [Chen *et al.*, 2011] proposed the co-training for domain adaptation that slowly adding to the training set both the target features and instances in which the current algorithm is the most confident. [Zou *et al.*, 2018] proposed the self-paced self-training (SPST) for unsupervised domain adaptation classification that can perform a self-paced learning [Tang *et al.*, 2012] with latent variable objective optimization. The representative SPST has inspired a number of follow-uppers such as [Zou *et al.*, 2021] and [Chen *et al.*, 2022b]. Nevertheless, all of existing self-training algorithms were designed for classification or segmentation, while self-training for the UDAVR remains a promising yet less explored direction.

3 Methodology

Given a source domain $\{\mathcal{I}_S^i, \boldsymbol{y}_S^i\}_{i=1}^{N_S}$ with N_S labeled samples and a target domain $\{\mathcal{I}_T^i\}_{i=1}^{N_T}$ with N_T unlabeled sam-

ples, tasks of UDAVR aim at learning a domain-invariant regression mapping to a shared continuous label space $\mathcal{Y}$. In the context of our focused 6D object pose estimation, the source and target domains are often the synthetic and real-world data, respectively, while the shared label space between two domains is the whole learning space of $SE(3)$.

To deal with the problems of UDAVR introduced in Sec. 1, *e.g.* , cumulative dependent nature and piece-wise manifolds in target space, we propose in this paper a manifold-aware self-training scheme, which decomposes the learning of $SE(3)$ space into a global classification on discretized pose anchors and a local pose refinement for feature scaling robustness, and incorporates a self-supervised manifold regularization to the self-training.

3.1 The Design of Network Architecture

Given a batch of B object-centric RGB images $\{\mathcal{I}^b\}_{b=1}^B$ as input, the proposed scheme is designed to predict 6D object poses $\{\mathcal{T}^b\}_{b=1}^B$, with each pose $\mathcal{T} = [\boldsymbol{R}|\boldsymbol{t}]$ represented by a 3D rotation $\boldsymbol{R} \in SO(3)$ and a 3D translation $\boldsymbol{t} \in \mathbb{R}^3$. The whole network architecture is shown in Fig. 2, which consists of three main modules, including a **Feature Extractor**, a **Coarse Classifier** of discretized pose anchors, and a **Fine Regressor of Residual Poses** to the nearest anchor.

More specifically, we employ the same feature extractor as [Labbé *et al.*, 2020] to learn the pose-sensitive feature vectors $\{\boldsymbol{f}^b \in \mathbb{R}^C\}_{b=1}^B$ from each frame, which are then fed into the decoupled coarse classifier and fine regressor individually, whose output are combined together as final pose predictions. The former learns coarse poses via classification on the discretized pose anchors, while the latter learns residual poses to refine the coarse ones of pose anchors locally; both modules share the same input features, achieving superior robustness against inconsistent feature scaling. We will take a single image as an example to detail the two pose estimation modules shortly, and thus omit the superscript b of the notations for simplicity in the following subsections.

Coarse Classification on Discretized Pose Anchors. Given the pose-sensitive feature $\boldsymbol{f}$ of $\mathcal{I}$, the goal of this module is to globally make coarse predictions of $\boldsymbol{R}$ and $\boldsymbol{t}$ via classification on their pre-defined anchors, respectively. For the rotation $\boldsymbol{R}$, we generate $N_{\boldsymbol{R}}$ anchors that are uniformly distributed on the whole $SO(3)$ space as [Li *et al.*, 2018a], which are denoted as $\{\boldsymbol{R}_a^1, \cdots, \boldsymbol{R}_a^{N_{\boldsymbol{R}}}\}$. For the translation $\boldsymbol{t}$, we factorize it into three individual classification targets, including the two translation components v_x and v_y on the image coordinate system along X-axis and Y-axis, with the remaining component z along Z-axis; for each classification target $t \in \{v_x, v_y, z\}$, we discretize the range of $[d_t^{min}, d_t^{max}]$ into N_t bins uniformly, and use the bin centers $\{t_a^1, \cdots, t_a^{N_t}\}$ as the anchors of t. We implement the classifier as four Multilayer Perceptrons (MLPs) with $N_{\boldsymbol{R}}, N_{v_x}, N_{v_y}, N_z$ output neurons, which are collectively denoted as the probabilities $\boldsymbol{S}_{\boldsymbol{R}} \in \mathbb{R}^{N_{\boldsymbol{R}}}, \boldsymbol{S}_{v_x} \in \mathbb{R}^{N_{v_x}}, \boldsymbol{S}_{v_y} \in \mathbb{R}^{N_{v_y}}$, and $\boldsymbol{S}_z \in \mathbb{R}^{N_z}$ of $\boldsymbol{R}, v_x, v_y$ and z, respectively. Denoting their indexes of maximal probabilities as $i_{\boldsymbol{R}}^{max}, i_{v_x}^{max}, i_{v_y}^{max}$ and i_z^{max}, the classifier finally gives out their coarse pose predictions as $\boldsymbol{R}_{cls} = \boldsymbol{R}_a^{i_{\boldsymbol{R}}^{max}}, v_{x,cls} = v_{x,a}^{i_{v_x}^{max}}, v_{y,cls} = v_{y,a}^{i_{v_y}^{max}}$ and $z_{cls} = z_a^{i_z^{max}}$.
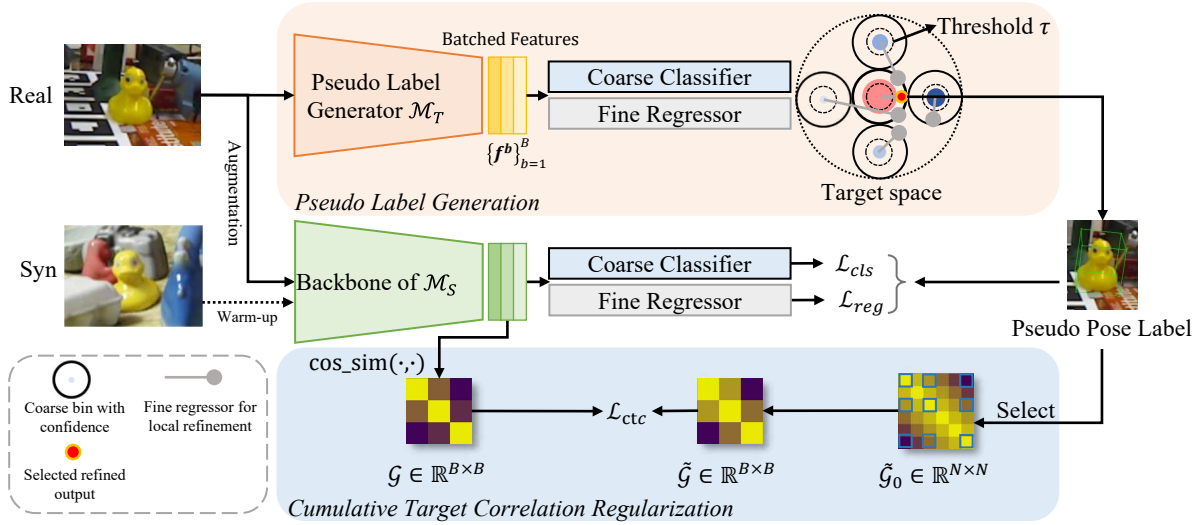
Figure 2: The pipeline of our manifold-aware self-training scheme.

Fine Regressor of Residual Poses. This module shares the same input feature $\boldsymbol{f}$ as the coarse classifier to make the learning more robust to feature scale variations, and is implemented as four MLPs with $N_{\boldsymbol{R}} \times 6, N_{v_x}, N_{v_y}, N_z$ output neurons to regress the residuals of the pose anchors. We collectively denote the outputs as $\{\boldsymbol{R}^i_{reg,6D}\}^{N_{\boldsymbol{R}}}_{i=1}, \{v^i_{x,reg}\}^{N_{v_x}}_{i=1}$, $\{v^i_{y,reg}\}^{N_{v_y}}_{i=1}$, and $\{z^i_{reg}\}^{N_z}_{i=1}$; here we use the continuous 6D representations of rotation [Zhou $et\ al.$, 2019] as the regression target, which can be transformed into rotation matrices $\{\boldsymbol{R}^i_{reg}\}^{N_{\boldsymbol{R}}}_{i=1}$. According to probabilities of the classifier, the fine regressor refines the coarse predictions via the residuals $\boldsymbol{R}_{reg} = \boldsymbol{R}^{i^{max}_{\boldsymbol{R}}}_{reg}$, $v_{x,reg} = v^{i^{max}_{v_x}}_{x,reg}$, $v_{y,reg} = v^{i^{max}_{v_y}}_{y,reg}$, and $z_{reg} = z^{i^{max}_z}_{reg}$.

Combining coarse anchor predictions and their residuals, our proposed network can generate the final object pose $\mathcal{T} = [\boldsymbol{R}|\boldsymbol{t}]$, with $\boldsymbol{t} = [x, y, z]$, as follows:

$$\begin{cases} \boldsymbol{R} = \boldsymbol{R}_{reg} \cdot \boldsymbol{R}_{cls} \\ x = (v_{x,cls} + v_{x,reg}) \cdot z/f_x \\ y = (v_{y,cls} + v_{y,reg}) \cdot z/f_y \\ z = z_{cls} + z_{reg} \end{cases}, \quad (1)$$

where f_x and f_y are the focal lengths along X-axis and Y-axis, respectively.

3.2 Manifold-Aware Objective

To train our network, we formulate the following manifold-aware objective $\mathcal{L}$ via combining a **coarse-to-fine pose decomposition loss** $\mathcal{L}_{pose}$ with a **cumulative target correlation regularization** $\mathcal{L}_{ctc}$:

$$\mathcal{L} = \mathcal{L}_{pose} + \mathcal{L}_{ctc}, \quad (2)$$

where $\mathcal{L}_{pose}$ favors for domain-invariant representations in 6D pose estimation across domains, while $\mathcal{L}_{ctc}$ enforces target manifolds into representation learning.

Coarse-to-fine Pose Decomposition Loss. $\mathcal{L}_{pose}$ consists of two loss terms $\mathcal{L}_{cls}$ and $\mathcal{L}_{reg}$ for the coarse classifier and the fine regressor, respectively, as follows:

$$\mathcal{L}_{pose} = \frac{1}{B} \sum_{b=1}^{B} \mathcal{L}^b_{cls} + \mathcal{L}^b_{reg}. \quad (3)$$

For simplicity, we introduce $\mathcal{L}_{pose}$ on single input, and thus omit the batch index b accordingly.

For the coarse classifier, given the ground truth pose $\tilde{\mathcal{T}} = [\tilde{\boldsymbol{R}}|\tilde{\boldsymbol{t}}]$, with $\tilde{\boldsymbol{t}} = [\tilde{x}, \tilde{y}, \tilde{z}]$ (and $\tilde{v}_x, \tilde{v}_y$), we first adopt a sparse scoring strategy to assign the labels for $\boldsymbol{S}_{\boldsymbol{R}}, \boldsymbol{S}_{v_x}, \boldsymbol{S}_{v_y}$ and $\boldsymbol{S}_z$, resulting in $\tilde{\boldsymbol{S}}_{\boldsymbol{R}}, \tilde{\boldsymbol{S}}_{v_x}, \tilde{\boldsymbol{S}}_{v_y}$ and $\tilde{\boldsymbol{S}}_z$, respectively, with each element $\tilde{s}^i_t$ ($t \in \{\boldsymbol{R}, v_x, v_y, z\}$) assigned as follows:

$$\tilde{s}^i_t = \begin{cases} \theta_{t,1}, & i \in \text{NN}_1(\tilde{t}) \\ \theta_{t,2}, & i \in \text{NN}_{k_t}(\tilde{t}) \backslash \text{NN}_1(\tilde{t}) \\ 0, & Otherwise \end{cases}, \quad (4)$$

where $\theta_{t,1} \gg \theta_{t,2}$, and $\theta_{t,1} + (k_t - 1)\theta_{t,2} = 1$. $\text{NN}_{k_t}(\tilde{t})$ denotes the set of indexes of the k_t nearest anchors of $\tilde{t}$.[1] With the assigned labels, we use the cross-entropy loss $\mathcal{H}$ on top of the classifier as follows:

$$\mathcal{L}_{cls} = \sum_{t \in \{\boldsymbol{R}, v_x, v_y, z\}} \mathcal{H}(\boldsymbol{S}_t, \tilde{\boldsymbol{S}}_t). \quad (5)$$

For the fine regressor, we make individual predictions on each anchor of $t \in \{\boldsymbol{R}, (v_x, v_y), z\}$ by combining the paired classification and regression results, and supervise the predic-

---

[1] We use the geodesic distance [Gao $et\ al.$, 2018] to measure the distance of two rotations $\boldsymbol{R}_1$ and $\boldsymbol{R}_2$ as $\arccos(\frac{trace(\boldsymbol{R}_1 \boldsymbol{R}_2^T)-1}{2})$, and use the difference value to measure that of two scalars.

tions of their top K nearest anchors of $\tilde{t}$ as follows:

$$
\begin{aligned}
\mathcal{L}_{reg} = & \sum_{i \in \mathrm{NN}_{k_{\boldsymbol{R}}}(\tilde{\boldsymbol{R}})} \mathcal{D}(\mathcal{T}_{\boldsymbol{R}^i}, \tilde{\mathcal{T}}) + \sum_{i \in \mathrm{NN}_{k_z}(\tilde{z})} \mathcal{D}(\mathcal{T}_{z^i}, \tilde{\mathcal{T}}) \\
& + \sum_{i \in \mathrm{NN}_{k_{v_x v_y}}(\tilde{v}_x \tilde{v}_y)} \mathcal{D}(\mathcal{T}_{v_x^i v_y^i}, \tilde{\mathcal{T}}),
\end{aligned}
\tag{6}
$$

where t^i denotes the prediction of the anchor i of t, and $\mathcal{T}_{t^i}$ denotes the object pose computed by t^i and other ground truths $\{\tilde{\boldsymbol{R}}, (\tilde{v}_x, \tilde{v}_y), \tilde{z}\} \backslash \tilde{t}$. $\mathcal{D}(\cdot, \cdot)$ is the L_1 distance between the point sets transformed by two object poses from the same object point cloud $\mathcal{O}$, as follows:

$$
\mathcal{D}(\mathcal{T}, \tilde{\mathcal{T}}) = \frac{1}{|\mathcal{O}|} \sum_{x \in \mathcal{O}} \|\mathcal{T}x - \tilde{\mathcal{T}}x\|_1.
\tag{7}
$$

Following [Labbé *et al.*, 2020], we combine the supervision of v_x and v_y for convenience in (6), and also employ the same strategy to handle object symmetries by finding the closest ground truth rotation to the predicted one.

Cumulative Target Correlation Regularization. For regression tasks, continuous targets preserve latent cumulative dependency [Chen *et al.*, 2013]. When we discretize the continuously changing targets into discretized labels as classification, the assumption of independence across targets is adopted, which is invalid in regressing continuous targets. As a result, each class cannot seek support from samples of correlated class, which can significantly reduce performance especially for sparse and imbalanced data distributions. To better cope with this problem, we propose to regularize the features by an explicit relation in the regression target space.

Given the pose-sensitive feature vectors $\{\boldsymbol{f}^b \in \mathbb{R}^C\}_{b=1}^B$ of a mini-batch inputs $\{\mathcal{I}^b\}_{b=1}^B$, we first build the feature correlation graph $\mathcal{G} \in \mathbb{R}^{B \times B}$ across the data batch via feature cosine similarities, with the element g^{ij} indexed by (i, j) computed as follows:

$$
g^{ij} = \frac{< \boldsymbol{f}^i, \boldsymbol{f}^j >}{\|\boldsymbol{f}^i\|_2 \cdot \|\boldsymbol{f}^j\|_2},
\tag{8}
$$

where $< \cdot, \cdot >$ denotes inner product. We then build the ground truth $\tilde{\mathcal{G}}$ based on a pre-computed correlation graph $\tilde{\mathcal{G}}_0 \in \mathbb{R}^{N \times N}$ with N pose classes; assuming the classes of $\mathcal{I}_i$ and $\mathcal{I}_j$ are n_i and n_j, respectively, we assign the value of $\tilde{g}^{ij} \in \tilde{\mathcal{G}}$ as that of $\tilde{g}_0^{n_i n_j}$. Finally, the proposed target correlation regularizer can be simply written as the squared L_2 distance between $\mathcal{G}$ and $\tilde{\mathcal{G}}$:

$$
\mathcal{L}_{ctc} = \|\mathcal{G} - \tilde{\mathcal{G}}\|_2^2.
\tag{9}
$$

There are multiple choices for building the pose-related correlation graph $\tilde{\mathcal{G}}_0$; here we introduce a simple but effective one, which utilizes the similarity of depth components of translations along Z-axis to initialize $\tilde{\mathcal{G}}_0$, with $N = N_z$. Specifically, for the anchors $\{z_a^1, \cdots, z_a^N\}$ of z, we map them linearly to the angles $\{\phi^1, \cdots, \phi^N\}$ as follows:

$$
\phi^n = \frac{z_a^n}{z^{max} - z^{min}} \cdot \frac{\pi}{2},
\tag{10}
$$

and the element $\tilde{g}_0^{n_i n_j}$ of $\tilde{\mathcal{G}}_0$ indexed by (n_i, n_j) can be defined as the cosine of difference between the angles:

$$
\tilde{g}_0^{n_i n_j} = \cos(|\phi^{n_i} - \phi^{n_j}|).
\tag{11}
$$

When $z_a^{n_i}$ and $z_a^{n_j}$ are close, the difference of their corresponding angles is small, and thus the correlation value of $\tilde{g}_0^{n_i n_j}$ will be large. The reason for choosing z is that the learning of this component is very challenging in 6D pose estimation without depth information. Experimental results in Sec. 4.2 also verify the effectiveness of our regularization.

3.3 Manifold-Aware Self-training

To reduce the Sim2Real domain gap, we design a manifold-aware self-training scheme for unsupervisedly adapting the pose estimator, which adaptively incorporates our proposed manifold-aware training objective in (2) with Self-Paced Self-Training [Zou *et al.*, 2018] to select target samples in an easy-to-hard manner. More specifically, we first train a teacher model $\mathcal{M}_T$ on the labeled synthetic data (source domain) as a pseudo-label annotator for the unlabeled real-world data (target domain), and select the training samples from the real data with pseudo labels for the learning of a student model $\mathcal{M}_S$. Both teacher and student models share the same networks introduced in Sec. 3.1, and are trained by solving the problems of $\min_{\mathcal{M}_T} \mathcal{L}$ and $\min_{\mathcal{M}_S} \mathcal{L}$, respectively.

The core of sample selection on the target domain lies on the qualities of pseudo labels. For the tasks of visual classification, the categorical probabilities are usually used as the measurement of qualities, while for those of visual regression tasks, *e.g.* , object pose estimation in this paper, direct usage of the typical mean square error (MSE) can be less effective due to lack of directional constraints for adaptation. In geometric viewpoint, the surface of a super ball can have the same MSE distance to its origin, but the optimal regions of object surface for domain adaptation exist, which can be omitted by the MSE metric. Owing to the decomposition of object pose estimation into coarse classification and fine regression in our MAST scheme, we can flexibly exploit the classification scores to indicate the qualities of pseudo labels, since the coarse classification points out the overall direction of pose estimation. In practice, we use the probabilities $\boldsymbol{S}_z$ as confidence scores because UDA on classification can perform more stably and robustly, and set a threshold τ to select the samples with scores larger than τ for training $\mathcal{M}_S$. Larger score indicates higher quality of the pseudo label. Following [Zou *et al.*, 2018], the threshold τ is gradually decreased during training, realizing the learning in an easy-to-hard manner and making $\mathcal{M}_S$ generalized to harder target samples.

4 Experiments

Datasets and Settings. The LineMOD dataset [Hinterstoisser *et al.*, 2012] provides individual videos of 13 textureless objects, which are recorded in cluttered scenes with challenging lighting variations. For each object, we follow [Brachmann *et al.*, 2014] to use randomly sampled 15% of the sequence as the real-world training data of the target domain, and the remaining images are set aside for testing. The Occluded LineMOD dataset [Brachmann *et al.*, 2014] is a subset

Method	Ape	Bvise	Cam	Can	Cat	Drill	Duck	*Eggbox*	*Glue*	Holep	Iron	Lamp	Phone	Mean
							Data: syn (w/ GT)							
AAE [Sundermeyer *et al.*, 2020]	4.0	20.9	30.5	35.9	17.9	24.0	4.9	81.0	45.5	17.6	32.0	60.5	33.8	31.4
DSC-PoseNet [Yang *et al.*, 2021]	23.4	75.6	11.7	40.1	26.7	53.8	14.0	73.6	26.7	19.5	56.2	39.4	20.0	37.0
MHP [Manhardt *et al.*, 2019]	11.9	66.2	22.4	59.8	26.9	44.6	8.3	55.7	54.6	15.5	60.8	-	34.4	38.8
DPOD [Zakharov *et al.*, 2019]	35.1	59.4	15.5	48.8	28.1	59.3	25.6	51.2	34.6	17.7	84.7	45.0	20.9	40.5
SD-Pose [Li *et al.*, 2021]	54.0	76.4	50.2	81.2	71.0	64.2	54.0	**93.9**	**92.6**	24.0	77.0	82.6	53.7	67.3
Self6D++ [Wang *et al.*, 2021a]	50.9	**99.4**	**89.2**	97.2	79.9	**98.7**	24.6	81.1	81.2	41.9	**98.8**	**98.9**	64.3	77.4
MAR (ours)	68.6	97.4	79.4	**98.3**	87.1	94.2	**61.3**	82.0	87.1	**56.7**	94.3	92.3	68.8	**82.1**
							Data: syn (w/ GT) + real (w/ GT)							
DPOD [Zakharov *et al.*, 2019]	53.3	95.2	90.0	94.1	60.4	97.4	66.0	99.6	93.8	64.9	**99.8**	88.1	71.4	82.6
DSC-PoseNet [Yang *et al.*, 2021]	59.2	98.1	88.0	92.1	79.4	94.5	51.7	98.5	93.9	78.4	96.2	96.3	90.0	85.9
Self6D++ [Wang *et al.*, 2021a]	**85.0**	99.8	**96.5**	99.3	93.0	**100.0**	65.3	**99.9**	98.1	73.4	86.9	99.6	86.3	91.0
MAR (ours)	81.4	**99.9**	90.7	**99.6**	**94.6**	98.1	**85.5**	97.6	**98.5**	**89.2**	97.1	**99.7**	**96.0**	**94.5**
							Data: syn (w/ GT) + real (w/o GT)							
Self6D-RGB [Wang *et al.*, 2020]	0.0	10.1	3.1	0.0	0.0	7.5	0.1	33.0	0.2	0.0	5.9	20.7	2.4	6.4
DSC-PoseNet [Yang *et al.*, 2021]	35.9	83.1	51.5	61.0	45.0	68.0	27.6	89.2	52.5	26.4	56.3	68.7	46.3	54.7
Zhang *et al.* [Zhang *et al.*, 2021]	–	–	–	–	–	–	–	–	–	–	–	–	-	60.4
Sock *et al.* [Sock *et al.*, 2020]	37.6	78.6	65.5	65.6	52.5	48.8	35.1	89.2	64.5	41.5	80.9	70.7	60.5	60.6
Self6D++ [Wang *et al.*, 2021a]	**76.0**	91.6	**97.1**	**99.8**	85.6	**98.8**	56.5	91.0	92.2	35.4	**99.5**	97.4	**91.8**	85.6
MAST (ours)	73.5	**97.2**	80.8	98.6	**89.1**	93.9	**66.9**	95.3	95.4	69.8	95.5	98.6	79.1	**87.2**

Table 1: Comparative evaluation on the LineMOD dataset w.r.t. the Average Recall (%) of ADD(-S). Symmetric object classes are in italic. 'MAR' (manifold-aware regression) denotes our method without self-training.

Method	Occluded LineMOD									HomebrewedDB			
	Ape	Can	Cat	Drill	Duck	*Eggbox*	*Glue*	Holep	Mean	Bvise	Drill	Phone	Mean
						Data: syn (w/ GT)							
DPOD [Zakharov *et al.*, 2019]	2.3	4.0	1.2	7.2	10.5	4.4	12.9	7.5	6.3	52.9	37.8	7.3	32.7
CDPN [Li *et al.*, 2019]	20.0	15.1	16.4	22.2	5.0	36.1	27.9	24.0	20.8	–	–	–	–
SD-Pose [Li *et al.*, 2021]	21.5	56.7	17.0	44.4	27.6	42.8	45.2	21.6	34.6	–	–	–	–
SSD6D+Ref. [Manhardt *et al.*, 2018]	–	–	–	–	–	–	–	–	–	82.0	22.9	24.9	43.3
Self6D++ [Wang *et al.*, 2021a]	44.0	**83.9**	**49.1**	**88.5**	15.0	**33.9**	75.0	34.0	52.9	7.1	2.2	0.1	3.1
MAR (ours)	**44.9**	78.4	40.3	73.5	**47.9**	26.9	72.1	**58.0**	**55.3**	**92.6**	**91.5**	**80.0**	**88.0**
						Data: syn (w/ GT) + real (w/o GT)							
DSC-PoseNet [Yang *et al.*, 2021]	13.9	15.1	19.4	40.5	6.9	38.9	24.0	16.3	21.9	72.9	40.6	18.5	44.0
Sock *et al.* [Sock *et al.*, 2020]	12.0	27.5	12.0	20.5	23.0	25.1	27.0	35.0	22.8	57.3	46.6	41.5	52.0
Zhang *et al.* [Zhang *et al.*, 2021]	–	–	–	–	–	–	–	–	33.7	–	–	–	63.8
Self6D++ [Wang *et al.*, 2021a]	**57.7**	**95.0**	**52.6**	**90.5**	26.7	45.0	**87.1**	23.5	59.8	56.1	**97.7**	**85.1**	79.6
MAST (ours)	47.6	82.9	45.4	75.0	**53.7**	**48.2**	75.3	**63.0**	**61.4**	**93.8**	91.5	81.8	**89.0**

Table 2: Comparative evaluation on the Occluded LineMOD and HomebrewedDB datasets w.r.t. the Average Recall (%) of the ADD(-S). Symmetric object classes are in italic. 'MAR' (manifold-aware regression) denotes our method without self-training.

of the LineMOD with 8 different objects, which is formed by the images with severe object occlusions and self-occlusions. We follow [Wang *et al.*, 2021a] to split the training and test sets. The HomebrewedDB dataset [Kaskman *et al.*, 2019] provides newly captured test images of three objects in the LineMOD, including bvise, driller and phone. Following the Self-6D [Wang *et al.*, 2020], the second sequence of HomebrewedDB is used to test our models which are trained on the LineMOD, to evaluate the robustness of our method on different variations, *e.g.* , scene layouts and camera intrinsics. In the experiments, the above three real-world datasets are considered as the target domains, all of which share the same synthetic source domain. We employ the publicly available synthetic data provided by BOP challenge [Hodaň *et al.*, 2020] as the source data, which contains 50k images generated by physically-based rendering (PBR) [Hodaň *et al.*, 2019].

Evaluation Metrics. Following [Wang *et al.*, 2020], we employ the Average Distance of model points (ADD) [Hinterstoisser *et al.*, 2012] as the evaluation metric of the 6D poses for asymmetric objects, which measures the average devia-

tion of the model point set $\mathcal{O}$ transformed by the estimated pose $\mathcal{T} = [\boldsymbol{R}|\boldsymbol{t}]$ and that transformed by the ground-truth pose $\tilde{\mathcal{T}} = [\tilde{\boldsymbol{R}}|\tilde{\boldsymbol{t}}]$:

$$\mathcal{D}_{\text{ADD}}(\mathcal{T}, \tilde{\mathcal{T}}) = \frac{1}{|\mathcal{O}|} \sum_{\boldsymbol{x} \in \mathcal{O}} \|(\boldsymbol{R}\boldsymbol{x} + \boldsymbol{t}) - (\tilde{\boldsymbol{R}}\boldsymbol{x} + \tilde{\boldsymbol{t}})\|_2. \quad (12)$$

For symmetric objects, we employ the metric of Average Distance of the closest points (ADD-S) [Hodaň *et al.*, 2016]:

$$\mathcal{D}_{\text{ADD-S}}(\mathcal{T}, \tilde{\mathcal{T}}) = \frac{1}{|\mathcal{O}|} \sum_{\boldsymbol{x}_1 \in \mathcal{O}} \min_{\boldsymbol{x}_2 \in \mathcal{O}} \|(\boldsymbol{R}\boldsymbol{x}_1 + \boldsymbol{t}) - (\tilde{\boldsymbol{R}}\boldsymbol{x}_2 + \tilde{\boldsymbol{t}})\|_2. \quad (13)$$

Combining (12) and (13), we report the Average Recall (%) of ADD(-S) less than 10% of the object's diameter on all the three datasets.

Implementation Details. For object detection, we use Mask R-CNN [He *et al.*, 2017] trained purely on synthetic PBR images to generate the object bounding boxes for the target real data. For pose estimation, we set the numbers of anchors as $N_{\boldsymbol{R}} = 60, N_{v_x} = N_{v_y} = 20, N_z = 40$, and set

the ranges of v_x, v_y and z as $[d_{v_x}^{min}, d_{v_x}^{max}] = [d_{v_y}^{min}, d_{v_y}^{max}] = [-200, 200]$, and $[d_z^{min}, d_z^{max}] = [0.0, 2.0]$, respectively. To train our network, we choose $\theta_1^{\boldsymbol{R}} = 0.7, \theta_2^{\boldsymbol{R}} = 0.1$ and $k_{\boldsymbol{R}} = 4$ for rotation in (4), and also set $\theta_1^{v_x} = \theta_1^{v_y} = \theta_1^z = 0.55, \theta_2^{v_x} = \theta_2^{v_y} = \theta_2^z = 0.075$, and $k_{v_x} = k_{v_y} = k_z = 7$ for translation. Following the popular setting [Wang *et al.*, 2021a], we train individual networks for all the objects with the Adam optimizer [Kingma and Ba, 2014]. The teacher model $\mathcal{M}_T$ is firstly pre-trained on the synthetic images of all objects, and then fine-tuned on the single object, while the parameters of the student model $\mathcal{M}_S$ is initialized as those of $\mathcal{M}_T$; their initial learning rates are 3×10^{-4} and 3×10^{-5}, respectively. The training batch size is set as $B = 32$. We also include the same data augmentation as [Labbé *et al.*, 2020] during training.

4.1 Comparative Evaluation

We compare our method with the existing ones on three benchmarks for 6D object pose estimation with RGB images.

On the LineMOD, we conduct experiments under three settings of training data, including 1) labeled synthetic data, 2) labeled synthetic and real data, and 3) labeled synthetic data and unlabeled real data. Results of the first two settings are the lower and upper bounds of that of the last setting. We report qualitative results of comparative methods in Table 1, where our method outperforms its competitors by large margins under all the settings, *e.g.* , with the respective improvements of 4.7%, 3.5% and 1.6% over the state-of-the-art Self6D++ [Wang *et al.*, 2021a]. On the Occluded LineMOD and the HomebrewedDB, results are shown in Table 2, where our method consistently performs better than the existing ones on both datasets, demonstrating superior robustness of our method against occlusion and the generalization to new scenes and cameras.

4.2 Ablation Studies and Analyses

Effects of Decomposing into Coarse Classification and Fine Regression. We decompose the problem of UDA on estimating object poses into a coarse classification on discretized anchors and a residual regression. As shown in Table 3, for the models trained purely on synthetic data, the design of pose decomposition realizes 4.0% and 5.7% improvements on the LineMOD and the Occluded LineMOD, respectively, compared to direct regression of object poses, since the global classification eases the difficulty in learning along with feature-scaling robustness, and the local regression achieves pose refinement.

Effects of Cumulative Target Correlation Regularization. As shown in Table 3, $\mathcal{L}_{ctc}$ consistently improves the results under different settings across different datasets, *e.g.* , 5.6% improvement on the Occluded LineMOD for the model trained on synthetic data, which demonstrates the effectiveness of $\mathcal{L}_{ctc}$ on mining latent correlation across regression targets. We also visualize the feature distribution of an example via the t-SNE [Van der Maaten and Hinton, 2008] in Fig. 1, where, with $\mathcal{L}_{ctc}$, features assigned to different pose anchors preserve smooth and continuously changing nature of regression targets in the feature space.

Pose Estimator	$\mathcal{L}_{ctc}$	Method of UDA	Dataset	
			LM	LMO
Data: syn (w/ GT)				
Reg.	×	-	75.3	44.0
Cls. + Reg.	×	-	79.3	49.7
Cls. + Reg.	✓	-	**82.1**	**55.3**
Data: syn (w/ GT) + real (w/o GT)				
Cls. + Reg.	✓	RSD [Chen *et al.*, 2021]	83.9	55.0
Cls. + Reg.	×	Self-Training	85.6	60.1
Cls. + Reg.	✓	Self-Training	**86.5**	**61.4**

Table 3: Ablation studies on LineMOD (LM) and Occluded LineMOD (LMO) datasets w.r.t. the Average Recall (%) of ADD(-S). 'Reg.' denotes direct regression of object poses, while 'Cls. + Reg.' denotes the combined use of coarse classification and fine regressor for pose estimation.

Effects of Manifold-Aware Self-Training on Coarse Classification. The self-training schemes have been verified their effectiveness on reducing the Sim2Real domain gap by incorporating the unlabeled real data into training via pseudo label generation and training sample selection. Taking our network with $\mathcal{L}_{ctc}$ as example, the results are improved from 55.3% to 61.4% on the Occluded LineMOD via self-training. Compared to the RSD [Chen *et al.*, 2021] designed for the problem of UDA on regression, our MAST scheme can significantly beat the competing RSD (see results in Table 3), where the only difference lies in replacing self-training on coarse classification with RSD on whole regression. Such an observation can again confirm the superiority of the proposed MAST scheme, consistently outperforming the state-of-the-art UDA on regression.

5 Conclusion

This paper proposes a novel and generic manifold-aware self-training scheme for UDA on regression, which is applied to the challenging 6D pose estimation of object instances. We address the UDAVR problem via decomposing it into coarse classification and fine regression, together with a cumulative target correlation regularization. Experiment results on three popular benchmarks can verify the effectiveness of our MAST scheme, outperforming the state-of-the-art methods with significant margins. It is worth pointing out that our MAST scheme can readily be applied to any UDA regression tasks, as the UDA on coarse classification making our method robust against feature scaling while maintaining latent cumulative correlation underlying in regression target space.

Acknowledgments

References

[Bao *et al.*, 2022] Yiwei Bao, Yunfei Liu, Haofei Wang, and Feng Lu. Generalizing gaze estimation with rotation consistency. In *CVPR*, 2022.

[Brachmann *et al.*, 2014] Eric Brachmann, Alexander Krull, Frank Michel, Stefan Gumhold, Jamie Shotton, and Carsten Rother. Learning 6d object pose estimation using 3d object coordinates. In *ECCV*, 2014.

[Chen *et al.*, 2011] Minmin Chen, Kilian Q Weinberger, and John Blitzer. Co-training for domain adaptation. *NeurIPS*, 2011.

[Chen *et al.*, 2013] Ke Chen, Shaogang Gong, Tao Xiang, and Chen Change Loy. Cumulative attribute space for age and crowd density estimation. In *CVPR*, 2013.

[Chen *et al.*, 2017] Xiaozhi Chen, Huimin Ma, Ji Wan, Bo Li, and Tian Xia. Multi-view 3d object detection network for autonomous driving. In *CVPR*, 2017.

[Chen *et al.*, 2021] Xinyang Chen, Sinan Wang, Jianmin Wang, and Mingsheng Long. Representation subspace distance for domain adaptation regression. In *ICML*, 2021.

[Chen *et al.*, 2022a] Hansheng Chen, Pichao Wang, Fan Wang, Wei Tian, Lu Xiong, and Hao Li. Epro-pnp: Generalized end-to-end probabilistic perspective-n-points for monocular object pose estimation. In *CVPR*, 2022.

[Chen *et al.*, 2022b] Yongwei Chen, Zihao Wang, Longkun Zou, Ke Chen, and Kui Jia. Quasi-balanced self-training on noise-aware synthesis of object point clouds for closing domain gap. In *ECCV*, 2022.

[Collet *et al.*, 2011] Alvaro Collet, Manuel Martinez, and Siddhartha S Srinivasa. The moped framework: Object recognition and pose estimation for manipulation. *The international journal of robotics research*, 2011.

[Fischler and Bolles, 1981] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 1981.

[Gao *et al.*, 2018] Ge Gao, Mikko Lauri, Jianwei Zhang, and Simone Frintrop. Occlusion resistant object rotation regression from point cloud segments. In *ECCV Workshops*, 2018.

[Geiger *et al.*, 2012] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *CVPR*, 2012.

[Gopalan *et al.*, 2011] Raghuraman Gopalan, Ruonan Li, and Rama Chellappa. Domain adaptation for object recognition: An unsupervised approach. In *ICCV*, 2011.

[He *et al.*, 2017] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *ICCV*, 2017.

[He *et al.*, 2020] Yisheng He, Wei Sun, Haibin Huang, Jianran Liu, Haoqiang Fan, and Jian Sun. Pvn3d: A deep point-wise 3d keypoints voting network for 6dof pose estimation. In *CVPR*, 2020.

[He *et al.*, 2021] Yisheng He, Haibin Huang, Haoqiang Fan, Qifeng Chen, and Jian Sun. Ffb6d: A full flow bidirectional fusion network for 6d pose estimation. In *CVPR*, 2021.

[Hinterstoisser *et al.*, 2012] Stefan Hinterstoisser, Vincent Lepetit, Slobodan Ilic, Stefan Holzer, Gary Bradski, Kurt Konolige, and Nassir Navab. Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes. In *ACCV*, 2012.

[Hodaň *et al.*, 2016] Tomáš Hodaň, Jiří Matas, and Štěpán Obdržálek. On evaluation of 6d object pose estimation. In *ECCV*, 2016.

[Hodaň *et al.*, 2019] Tomáš Hodaň, Vibhav Vineet, Ran Gal, Emanuel Shalev, Jon Hanzelka, Treb Connell, Pedro Urbina, Sudipta N Sinha, and Brian Guenter. Photorealistic image synthesis for object instance detection. In *ICIP*, 2019.

[Hodaň *et al.*, 2020] Tomáš Hodaň, Martin Sundermeyer, Bertram Drost, Yann Labbé, Eric Brachmann, Frank Michel, Carsten Rother, and Jiří Matas. BOP challenge 2020 on 6D object localization. *ECCV Workshops*, 2020.

[Kaskman *et al.*, 2019] Roman Kaskman, Sergey Zakharov, Ivan Shugurov, and Slobodan Ilic. Homebreweddb: Rgb-d dataset for 6d pose estimation of 3d objects. In *ICCV Workshops*, 2019.

[Kehl *et al.*, 2017] Wadim Kehl, Fabian Manhardt, Federico Tombari, Slobodan Ilic, and Nassir Navab. Ssd-6d: Making rgb-based 3d detection and 6d pose estimation great again. In *ICCV*, 2017.

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

[Labbé *et al.*, 2020] Yann Labbé, Justin Carpentier, Mathieu Aubry, and Josef Sivic. Cosypose: Consistent multi-view multi-object 6d pose estimation. In *ECCV*, 2020.

[Lee and others, 2013] Dong-Hyun Lee et al. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, 2013.

[Lee *et al.*, 2022] 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 *CVPR*, 2022.

[Li *et al.*, 2018a] Chi Li, Jin Bai, and Gregory D Hager. A unified framework for multi-view multi-class object pose estimation. In *ECCV*, 2018.

[Li *et al.*, 2018b] Yi Li, Gu Wang, Xiangyang Ji, Yu Xiang, and Dieter Fox. Deepim: Deep iterative matching for 6d pose estimation. In *ECCV*, 2018.

[Li *et al.*, 2019] Zhigang Li, Gu Wang, and Xiangyang Ji. Cdpn: Coordinates-based disentangled pose network for real-time rgb-based 6-dof object pose estimation. In *ICCV*, 2019.

[Li *et al.*, 2021] Zhigang Li, Yinlin Hu, Mathieu Salzmann, and Xiangyang Ji. Sd-pose: Semantic decomposition for cross-domain 6d object pose estimation. In *AAAI*, 2021.

[Lin *et al.*, 2022] 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 *ECCV*, 2022.

[Manhardt *et al.*, 2018] Fabian Manhardt, Wadim Kehl, Nassir Navab, and Federico Tombari. Deep model-based 6d pose refinement in rgb. In *ECCV*, 2018.

[Manhardt *et al.*, 2019] Fabian Manhardt, Diego Martin Arroyo, Christian Rupprecht, Benjamin Busam, Tolga Birdal, Nassir Navab, and Federico Tombari. Explaining the ambiguity of object detection and 6d pose from visual data. In *ICCV*, 2019.

[Marchand *et al.*, 2015] Eric Marchand, Hideaki Uchiyama, and Fabien Spindler. Pose estimation for augmented reality: a hands-on survey. *IEEE transact. on visualization and computer graphics*, 2015.

[Park *et al.*, 2019] Kiru Park, Timothy Patten, and Markus Vincze. Pix2pose: Pixel-wise coordinate regression of objects for 6d pose estimation. In *ICCV*, 2019.

[Peng *et al.*, 2019] Sida Peng, Yuan Liu, Qixing Huang, Xiaowei Zhou, and Hujun Bao. Pvnet: Pixel-wise voting network for 6dof pose estimation. In *CVPR*, 2019.

[Roy *et al.*, 2019] Subhankar Roy, Aliaksandr Siarohin, Enver Sanigeto, Samuel Rota Bulo, Nicu Sebe, and Elisa Ricci. Unsupervised domain adaptation using feature-whitening and consensus loss. In *CVPR*, 2019.

[Sock *et al.*, 2020] Juil Sock, Guillermo Garcia-Hernando, Anil Armagan, and Tae-Kyun Kim. Introducing pose consistency and warp-alignment for self-supervised 6d object pose estimation in color images. In *3DV*, 2020.

[Sohn *et al.*, 2020] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *NeurIPS*, 2020.

[Su *et al.*, 2015] Hao Su, Charles R Qi, Yangyan Li, and Leonidas J Guibas. Render for cnn: Viewpoint estimation in images using cnns trained with rendered 3d model views. In *ICCV*, 2015.

[Sundermeyer *et al.*, 2020] Martin Sundermeyer, Zoltan-Csaba Marton, Maximilian Durner, and Rudolph Triebel. Augmented autoencoders: Implicit 3d orientation learning for 6d object detection. *IJCV*, 2020.

[Tang *et al.*, 2012] Kevin Tang, Vignesh Ramanathan, Li Fei-Fei, and Daphne Koller. Shifting weights: Adapting object detectors from image to video. *NeurIPS*, 2012.

[Tekin *et al.*, 2018] Bugra Tekin, Sudipta N Sinha, and Pascal Fua. Real-time seamless single shot 6d object pose prediction. In *CVPR*, 2018.

[Van der Maaten and Hinton, 2008] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 2008.

[Wang *et al.*, 2020] Gu Wang, Fabian Manhardt, Jianzhun Shao, Xiangyang Ji, Nassir Navab, and Federico Tombari. Self6d: Self-supervised monocular 6d object pose estimation. In *ECCV*, 2020.

[Wang *et al.*, 2021a] Gu Wang, Fabian Manhardt, Xingyu Liu, Xiangyang Ji, and Federico Tombari. Occlusion-aware self-supervised monocular 6d object pose estimation. *IEEE TPAMI*, 2021.

[Wang *et al.*, 2021b] Gu Wang, Fabian Manhardt, Federico Tombari, and Xiangyang Ji. Gdr-net: Geometry-guided direct regression network for monocular 6d object pose estimation. In *CVPR*, 2021.

[Xiang *et al.*, 2018] Yu Xiang, Tanner Schmidt, Venkatraman Narayanan, and Dieter Fox. Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes. *RSS*, 2018.

[Xie *et al.*, 2020] Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V Le. Self-training with noisy student improves imagenet classification. In *CVPR*, 2020.

[Xu *et al.*, 2019] Zelin Xu, Ke Chen, and Kui Jia. W-posenet: Dense correspondence regularized pixel pair pose regression. *arXiv preprint arXiv:1912.11888*, 2019.

[Xu *et al.*, 2022] Yan Xu, Kwan-Yee Lin, Guofeng Zhang, Xiaogang Wang, and Hongsheng Li. Rnnpose: Recurrent 6-dof object pose refinement with robust correspondence field estimation and pose optimization. In *CVPR*, 2022.

[Yang *et al.*, 2021] Zongxin Yang, Xin Yu, and Yi Yang. Dsc-posenet: Learning 6dof object pose estimation via dual-scale consistency. In *CVPR*, 2021.

[Yue *et al.*, 2021] Xiangyu Yue, Zangwei Zheng, Shanghang Zhang, Yang Gao, Trevor Darrell, Kurt Keutzer, and Alberto Sangiovanni Vincentelli. Prototypical cross-domain self-supervised learning for few-shot unsupervised domain adaptation. In *CVPR*, 2021.

[Zakharov *et al.*, 2019] Sergey Zakharov, Ivan Shugurov, and Slobodan Ilic. Dpod: 6d pose object detector and refiner. In *ICCV*, 2019.

[Zhang *et al.*, 2021] Shaobo Zhang, Wanqing Zhao, Ziyu Guan, Xianlin Peng, and Jinye Peng. Keypoint-graph-driven learning framework for object pose estimation. In *CVPR*, 2021.

[Zhou *et al.*, 2019] Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. On the continuity of rotation representations in neural networks. In *CVPR*, 2019.

[Zou *et al.*, 2018] Yang Zou, Zhiding Yu, BVK Kumar, and Jinsong Wang. Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In *ECCV*, 2018.

[Zou *et al.*, 2021] Longkun Zou, Hui Tang, Ke Chen, and Kui Jia. Geometry-aware self-training for unsupervised domain adaptation on object point clouds. In *ICCV*, 2021.