GS-Pose: Category-Level Object Pose Estimation via Geometric and Semantic Correspondence

Pengyuan Wang[1], Takuya Ikeda[2], Robert Lee[2], and Koichi Nishiwaki[2]

[1] Technical University of Munich Germany
pengyuan.wang@tum.de
[2] Woven by Toyota, Japan
{takuya.ikeda,robert.lee, koichi.nishiwaki}@woven.toyota
https://woven-planet.github.io/GS-Pose

Abstract. Category-level pose estimation is a challenging task with many potential applications in computer vision and robotics. Recently, deep-learning-based approaches have made great progress, but are typically hindered by the need for large datasets of either pose-labelled real images or carefully tuned photorealistic simulators. This can be avoided by using only geometry inputs such as depth images to reduce the domain-gap but these approaches suffer from a lack of semantic information, which can be vital in the pose estimation problem. To resolve this conflict, we propose to utilize both geometric and semantic features obtained from a pre-trained foundation model. Our approach projects 2D semantic features into object models as 3D semantic point clouds. Based on the novel 3D representation, we further propose a self-supervision pipeline, and match the fused semantic point clouds against their synthetic rendered partial observations from synthetic object models. The learned knowledge from synthetic data generalizes to observations of unseen objects in the real scenes, without any fine-tuning. We demonstrate this with a rich evaluation on the NOCS, Wild6D and SUN RGB-D benchmarks, showing superior performance over geometric-only and semantic-only baselines with significantly fewer training objects.

1 Introduction

Object pose estimation is a fundamental problem in the computer vision and robotics fields. With the advancement of deep learning methods, various learning-based pose estimation approaches have proven effective for instance-level pose estimation [19, 23, 39, 53, 61, 67]. Furthermore, recent approaches have extended the pose estimation problem from instance-level to category-level, estimating the pose of unseen object instances within a given category. However, most methods [4], [8], [9], [32] rely on the real dataset with annotated object poses, which are time-consuming to collect. To avoid heavy annotation efforts in the real dataset, methods [12], [5], [63], [26] train on synthetic object models and generalize it to real scenes. However, there exists domain gap between synthetic and real data including both RGB and depth images, because it is hard to fully

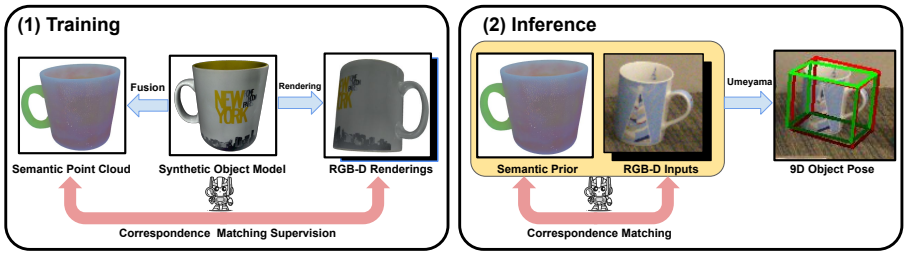


Fig. 1: We propose a novel self-supervision approach for category-level pose estimation that makes use of 3D semantic features from synthetic CAD models. (1) For synthetic object models, 2D semantic features are fused into their 3D point cloud. We then render RGB-D images of synthetic objects from different camera poses and train a matching network to learn from semantic point clouds and their rendered partial observations. (2) At inference time, we utilize the trained network to match a selected semantic prior against partial observations of novel objects, and then recover the object poses. Our approach is robust to the visual appearance of object instances and generalizes to novel objects in real scenes.

simulate the environment lighting, object texture, the sensor noise etc. The difference in the training and test data distributions leads to deteriorated network performances, which makes it hard to compete with supervised methods from the real data.

As depth information suffers less from domain gap than RGB information, methods [12], [63], [26] trained on synthetic data focus on geometry only, without making use of RGB information. This means the synthetic data needs to only cover the distribution of shape variety, not texture and color. However, relying on geometric information alone is not adequate to solve all ambiguities present in the pose estimation problem. For example, observing semantically meaningful parts of an object, such as the keyboard or display of a laptop, should help disambiguate the pose, even if the difference in geometry is minimal.

The challenges in utilizing color features in the synthetic training motivates us to rethink the problem from a different angle. Even though the synthetic RGB images are heavily affected by the different object instance textures and the domain gap, semantic information hidden in the images preserves the same for category-level instances and provides crucial guidance to disambiguate geometric matchings. To this end, we employs a pre-trained foundation model DINOv2 [38] to extract semantic features provided from 2D RGB images. To further lift the 2D semantic knowledge to the 3D point cloud, we sample multiple camera poses around the synthetic CAD model and project the 2D DINOv2 features to the point cloud based on the point visibility. Considering that points are visible from multiple views, we average the features from multiple observations as a feature per point. By this way, the object point cloud is enriched with semantic features per point which we name as a semantic point cloud as shown in Fig. 1.

Having embedded semantic features in the point cloud, another challenge is to estimate their correspondences with novel object instances from RGB-D inputs, without using any real training data. As an interesting observation, self-supervision methods [45], [37] generate a learning signal from the data itself for learning, without the need of external labels. Based on the idea, we train a matching network to estimate the correspondences between the semantic point cloud and their 2D RGB-D renderings in a self-supervised way. Firstly we generate partial and full semantic point cloud pairs from the RGB-D renderings and synthetic models, and obtain ground truth correspondences from the synthetic model itself. Next we combine semantic features and geometric features together in the training, to fuse both global and local information and boost their performance. Because semantic features instead of raw RGB images are leveraged for the synthetic training, our trained network generalizes to novel objects in the real scenes. To deal with symmetric categories, we re-align the ground truth correspondences as unique ones according to the symmetric axis in the training. At inference time, we select a synthetic model as semantic prior and match it to RGB-D inputs of novel instances without any fine-tuning. Through exhaustive evaluations on multiple real datasets, our self-supervision pipeline shows superior performances over other synthetic-only baselines and achieves competitive results in comparison with methods trained on the real annotated dataset. In summary, our proposed method features following contributions:

1. We fuse 2D semantic features to 3D semantic point clouds, and the combined global semantic and local geometric features result in a boosted network performance and domain generalization ability.
2. We propose a powerful self-supervision pipeline to match between the semantic point cloud and their RGB-D renderings from synthetic CAD models. The trained network generalizes to novel object instances in the real scenes, without the need of any real annotated dataset.
3. We conduct rich evaluations on multiple real datasets including NOCS [55], Wild6D [65], SUN RGB-D [48]. Exhaustive evaluations show that our simple yet effective approach greatly outperforms semantic-only or geometric-only baselines and have competitive results in comparison with methods trained on real data, while requiring as few as ten synthetic CAD models per category.

2 Related Work

2.1 Category-Level Object Pose Estimation

In the past few years, instance-level object pose estimation networks has made great progress in computer vision and robotics fields [11, 17, 18, 21–23, 27, 28, 40, 41, 44, 47, 49, 52, 54, 59, 60, 64]. Further, category-level object pose estimation networks are proposed to handle unseen object instances in the category without re-training [3, 4, 8, 9, 20, 30–32, 35, 50, 56–58, 63, 66].

Approaches Trained on Real Annotated Data Most category-level pose

methods are trained on the real annotated dataset which avoids the domain gap. NOCS [55] predicts normalized object coordinate space map from RGB images and then recover the object pose from depth images. Further methods [4, 8, 29, 31, 32] directly leverage geometric features from depth images for the training without using RGB images. Especially VI-Net [32] gets excellent results by decoupling rotations on the spherical representations. Instead of only regressing object poses, methods [3, 9, 20, 30, 35, 50, 56, 65] leverage shape priors and jointly estimate the input object shapes, which are important for robotic applications. For example DPDN [30] learns a shape prior deformation network in a self-supervised approach on the real data.To alleviate the real annotations needed for the training, methods [24, 25, 36, 57] firstly train on synthetic data and perform unsupervised learning on real scenes. However, collecting real images from a large amount of viewing angles is still needed.

Approaches Trained on Synthetic Data Real data with annotations is extremely hard to collect, considering that many object categories exist in the household environment and need to be supported for robotics applications. Methods [5, 12, 26, 62, 63] explore training with only synthetic object CAD models but try to generalize to real scenes. Gao et al. [12] uses partial object point cloud as inputs and refines object poses. Chen et al. [5] trains an implicit object renderer and optimizes object poses by novel view synthesis. CPPF [63] leverages the adapted point pair features and train a pose regression network on a large amount of category-level instances from ShapeNet [1]. The approaches mostly leverage geometric features, which fails to leverage the potential of RGB information. CPPF++ [62] further trains another network with RGB features and takes the advantage of ensemble models from two predictions, which however sacrifices the inference speed. Our method fuses semantic and geometric information in one network, and achieves superior performances with real-time capability.

2.2 Correspondences from Semantic Features

Color images are enriched with semantic features which guide the object pose estimations. Given an object CAD model, instance-level object pose estimation networks [64], [54] predict dense object correspondences from input color images and recover the 6D object poses from the 2D-to-3D correspondences. However, category-level objects have different shapes and appearances, which make the 2D-to-3D correspondence predictions more challenging. NOCS [55] proposes to learn the object shape and the correspondence matching jointly in a normalized object coordinate space on a large amount of rendered synthetic objects. Recently methods such as ZSP [15], [14] directly calculates correspondences from 2D-to-2D semantic features with multiple 2D object views in a zero-shot setting. As the drawback, the method needs to run on multiple object views up to 5 which is time-consuming. In contrast, our method only needs to inference once from the RGB-D input to the 3D template object and is real-time capable. Also directly estimating the correspondences from global semantic features are prone to outliers and our feature fusion method delivers more accurate result, which is shown in the experiments. Other 2D keypoint matching methods such

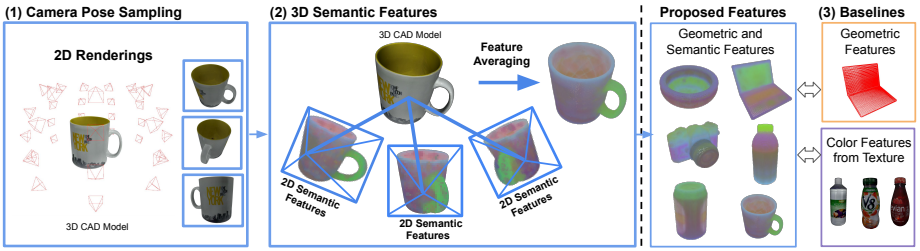


Fig. 2: Overview of semantic and geometric feature embedding. Different from other synthetic-only pose estimation pipelines, our method incorporates both geometric and semantic features to improve performance. (1) Firstly, we sample camera poses around the synthetic object CAD model with 2D RGB-D image renderings. (2) Afterwards, we fuse 2D semantic features from rendered RGB image to 3D point clouds as 3D semantic features. Specially, we project each point to the visible 2D observations and extract the 2D semantic feature on the projected image location. As an object point can be observed from multiple views, we calculate the average over the observed features and get a smooth representation. We directly use the 3D object point coordinates as geometric features and combine them with fused semantic features as the matching network inputs. (3) In comparison, baseline methods such as CPPF [63] only utilize geometric features, while others (NOCS [55]) leverage RGB images and need a large amount of textured objects for the training. In contrast, our network requires much fewer training objects with a good performance with the novel semantic representation in 3D space.

as SuperGlue [46] and LightGlue [34] firstly extract keypoint features from color images and fuse the features with a transformer network for the feature fusion. Especially LightGlue considers the keypoint inlier probability design which reduces potential outliers. Therefore we leverage LightGlue as our feature fusion backbone and modify the correspondence matching from 2D image keypoints to the 3D semantic point clouds.

2.3 Correspondences from Geometric Features

Recent advances of stereo and Time-of-Flight (ToF) cameras enable the depth perception of the environment. Therefore, geometric features from the projected point cloud are widely utilized in point cloud registrations. PPF-Net [7], PPF-FoldNet [6], GeoTransformer [43] firstly extracts point pair features (PPF) from the point cloud and leverages PointNet [42] or transformer for the matchings. CPPF [63] extends the PPF features to estimate the categoy-level objects. However, the above geometric features are limited to local features and fail to distinguish geometric parts which have similar shapes, for example the laptop lid and the keyboard. As a result, CPPF has a poor performance on the challenging mug and laptop category in comparison with bowls and bottles. By considering both global semantic feautres and local geometric features, our method shows superior performances even for the challenging categories.

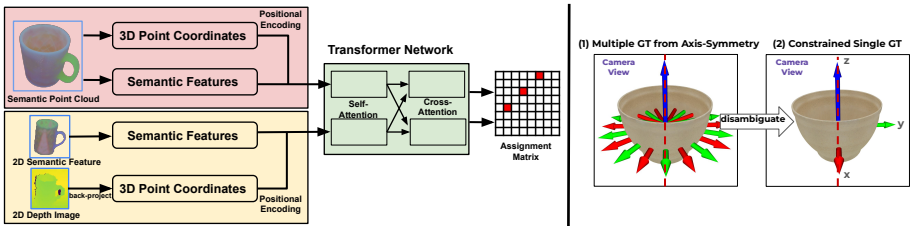


Fig. 3: Overview of our matching network. Left: For matching between a semantic point cloud and the RGB-D input, we firstly extract 2D semantic features from the RGB image and back-project the semantic features with the depth image as a partial input point cloud. We then uniformly sample 3000 points from the semantic point cloud and 1000 points from partial input point cloud for the matching. The normalized point coordinates are embedded as geometric features with positional encoding and added with semantic features. The embedded features are fused with self- and cross-attention layers for multiple iterations in a transformer network for global perceptions. The assignment matrix is calculated based on the cosine similarity of the fused point features. Right: To disambiguate the symmetrical poses, (1) Since multiple ground truth poses can exist for axis-symmetry objects, (2) the Ground Truth(GT) pose is constrained to intersect the object xz-plane with the camera origin coordinate system.

3 GS-Pose

Problem Definition We assume a limited amount of synthetic CAD models $S = \{S_i, |i = 1, \cdots, O\}$ are available in one category during training. Given a RGB-D image with detection mask of a novel instance for this category at inference time, our task is to recover the 9D object pose including the rotation $R \in SO(3)$, translation $t \in \mathbb{R}^3$ and scale $s \in \mathbb{R}^3$, assuming access to a single reference CAD model from the set S. No real images with pose annotations are available during training.

Overview As a category-level pose estimation method self-supervised from synthetic CAD models, GS-Pose estimates correspondences between the RGB-D input and synthetic models leveraging fused semantic and geometric features. For the training, 2D semantic features are lifted to 3D CAD models and generates semantic point clouds, which is visualized in Fig. 2. Afterwards, a matching network is trained self-supervised by matching the semantic point clouds against RGB-D renderings from their own synthetic CAD model, which is visualized in (1) from Fig. 1 and Fig. 3. In addition we deal with symmetric categories to avoid pose ambiguities. At inference time, we take a selected semantic shape prior and match it against RGB-D inputs of novel instances in the real scenes. The object poses are recovered from the matched correspondences and visualized in (2) from Fig. 1.

3.1 Semantic and Geometric Feature Embedding

Semantic Features on 2D Images Utilizing a semantic representation from a pre-trained foundation model would reduce the sensitivity to texture differ-

ences while providing vital global information to help tackle ambiguous geometry structures. Image foundation models, typically trained on web-scale data, provide a powerful base model, the features from which are able to reflect object semantic. With the prevalence of the transformer architecture, foundation models based on vision transformers such as DINOv2 [38] are able to better capture global relationships inside the features.

Semantic Features Lifting to 3D Point Cloud Despite powerful 2D foundation models, 3D foundation models for point clouds are still yet to be thoroughly explored, because of the challenge in collecting large scale 3D assets for training. To tackle this challenge, we reuse the 2D foundation models and project the features to the 3D object point cloud P to be used for 3D-3D matching. As shown in Fig. 2, we first sample camera poses $T_j, j \in \{1 \,..\, C\}$ around the objects, ensuring the model points are visible in at least one view. Next, the rendered RGB images are transformed to semantic features F_{2d} with DINOv2 which are later resized to the original image size of 480 x 480. For each frame j, the visibilities $V_{p,j}$ of object vertices p in the mesh are calculated. Based on the camera pose T and intrinsic K, the visible points are projected to the 2D feature image to retrieve the corresponding semantic features, as shown in Equ. 1. To align the feature discrepancies from multiple observations, we take the average of the visible features from multiple views for each point, as formulated in Equ. 2. The averaging additionally filters the noise from multiple predictions, shown in (2) from Fig. 2.

$$\forall(p_i \in P,\, j \in \{1 \,..\, C\}) \quad \begin{pmatrix} p_{ij,x} \\ p_{ij,y} \\ 1 \end{pmatrix} = K \cdot T_j^{-1} \cdot p_i \tag{1}$$

$$F_{p_i} = \frac{1}{\sum_{j=0}^{C} V_{p_i,j}} \sum_{j=0}^{C} V_{p_i,j} \cdot F_{2d,j}(p_{ij,x}, p_{ij,y}) \tag{2}$$

Geometric Features on 3D Point Cloud To gather geometric point features, a typical approach is to calculate the point pair features (PPF) based on their neighboring point distances and normals. However, recent networks [43] based on transformer architecture take point coordinates as inputs and use high frequency functions to embed geometric information of the points. In empirical experiments we find this approach effective in extracting local features from the point cloud. Therefore we directly take the point coordinates as geometric features and combine it with semantic features for the matching task.

3.2 Self-Supervision from Synthetic CAD Models

Motivation Embedding both semantic and geometric features in the given synthetic CAD models, we get semantic point clouds $P_s = \{(F_{p_i}, p_i),\, \forall p_i \in P\}$. Our goal is to estimate novel object poses (R, t, s) from their RGB-D inputs. To solve the problem, we take a matching-based approach which estimates the correspondences between the RGB-D inputs and the semantic point cloud P_s, and then recover the input object 6D poses with scales. The challenge is to avoid using

real training dataset with pose annotations, while delivering competitive results on the real test scenes. This motivates us to design a self-supervised pipeline fully exploiting the synthetic object models themselves, i.e. train the network based on the semantic point clouds P_s and their partial RGB-D observations from 2D renderings. With this approach, we only need semantic point cloud and their rendering pairs from the synthetic CAD models for the training and can generalize it to novel objects in real scenes.

Synthetic Training Pairs To collect semantic point clouds P_s and 2D renderings pairs for the training, we reuse the RGB-D renderings along with the extracted DINOv2 features from sampled object poses shown in (1) from Fig. 2. We then back-project depth images to an input point cloud Q, and then embed 2D semantics into a partial semantics object cloud $Q_s = \{(F_{q_i}, q_i), \forall q_i \in Q\}$. The partial point cloud Q_s is centered and normalized as training inputs, and the ground truth correspondences are retrieved by the finding the mutual nearest points inside the semantic point cloud P_s, given the rendering camera pose T. We use only 10 synthetic object models per category and 40 rendering camera poses per object.

Global and Local Feature Fusion Given two point clouds with semantic features P_s and Q_s, we fuse the semantic features and geometric features jointly in a transformer network. The semantic features provide high-level understandings of object parts as global information, while geometric features are enriched with detailed information of local geometries. The joint training from global and local information disambiguate object poses and boosts the matching performances, which is proved in exhaustive experiments against geometric-only or semantic-only baselines in the experiments. **Domain Generalization** The key design for domain generalization is to avoid the direct usage of raw RGB images. The semantic information is employed in the training for adjusting various possible object textures. Also the semantic features adapt well to real images thanks to the large scale pre-training. In addition depth information suffers less from the domain gap, which makes it possible to train the fusion network with only synthetic data. Therefore our synthetic training generalizes to novel objects in real scenes with high data-efficiency, as much fewer synthetic objects are required in comparison with geometric-only baselines [63]. **Disambiguating Symmetry** For many objects, there exist symmetries that cause ambiguities in the object pose, where the network will be trained against conflicting ground truth signals for a given pose. This presents a significant challenge in the pose estimation problem. Therefore, we extract unique ground truth poses by constraining the object xz-plane, (red and blue-axis plane as shown in the Fig. 3) to always intersect with the origin of camera coordinate system.We also treat the mug as an axis-symmetry object when the handle is invisible in the view.

3.3 Training Implementations

Matching Network Implementation Following by GeoTransformer [43], SuperGlue [34] and LightGlue [34], we utilize a transformer structure with multiple self- and cross-attention layers to fuse both semantic and geometric features of

two point clouds, as shown in Fig. 3. Specifically, the geometry features are embedded with the positional encodings of point coordinates and concatenated with the semantic features as network inputs. Matching between partial observations Q_s and the full 3D reference features P_s has an advantage of faster inference time than multiple 2D-2D partial matchings such as ZSP [15]. However, this may additionally increase the potential for mismatches to occur as the possible matching regions are also expanded. To avoid matching to regions that are out of input view visibility, we find it empirically useful to predict the inlier probability from LightGlue [34] in addition to the output features.

Training and Inference Assume the partial input point cloud Q has M points and the full object point cloud P has N points. After the transformer feature fusion, the fused features are F^Q and F^P with corresponding inlier probabilities as σ^P and σ^Q. The assignment matrix $\hat{A}$ is obtained by multiplication of the cosine similarity from F^Q and F^P, and the inlier probabilities, as recorded in Equ. 3. The training loss is the sum of inlier classification losses and the assignment matrix loss. The inlier classification losses are in Equ. 4 for partial inputs Q and Equ. 5 for full inputs P. The assignment matrix loss L_A is calculated in Equ. 6 with the focal loss [33] and γ as 2. A_{pos} and A_{neg} are the positive and negative ground truths for the assignment matrix A. Based on the assignment matrix from the output features, a threshold is applied to extract high confidence matches. At inference time, Umeyama algorithm [51] combined with RANSAC algorithm [10] is applied based on the matched correspondences with a selected shape prior to robustly recover the rotation, translation and the object scales.

$$\forall(i \in \{1..M\},\, j \in \{1..N\}) \quad \hat{A}_{i,j} = \sigma_i^P \cdot \sigma_j^Q \cdot A_{i,j} \tag{3}$$

$$L_Q = -\frac{1}{N}\sum_{j=0}^{M}(\sigma_{j,gt}^Q log\sigma_j^Q + (1-\sigma_{j,gt}^Q)log(1-\sigma_j^Q)) \tag{4}$$

$$L_P = -\frac{1}{M}\sum_{i=0}^{M}(\sigma_{i,gt}^P log\sigma_i^P + (1-\sigma_{i,gt}^P)log(1-\sigma_i^P)) \tag{5}$$

$$\begin{aligned} L_A = &-\frac{1}{|A_{pos}|}\sum_{\hat{A}_{i,j}\in A_{pos}}(1-\hat{A}_{i,j})^\gamma \log(\hat{A}_{i,j}) \\ &-\frac{1}{|A_{neg}|}\sum_{\hat{A}_{i,j}\in A_{neg}}\hat{A}_{i,j}^\gamma \log(1-\hat{A}_{i,j}) \end{aligned} \tag{6}$$

4 Experiments

4.1 Datasets

To cover as many categories and novel instances as possible, three datasets: NOCS [55], Wild6D [65] and SUN RGB-D [48] are employed for the evaluations. We train GS-Pose from ShapeNet [1] objects for the bottle, bowl, camera, can,

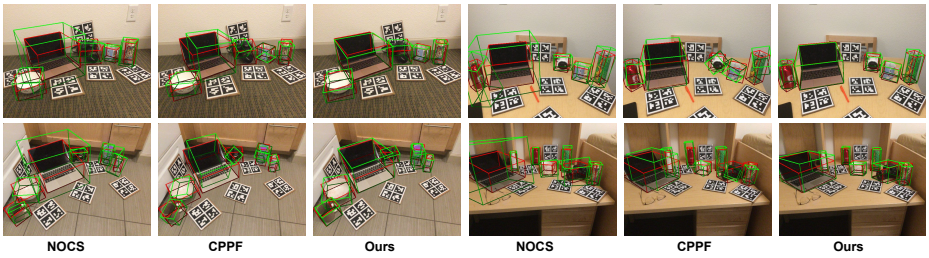
lpatop, mug categories and evaluate on the NOCS REAL275 dataset and Wild6D dataset. To test on challenging scenes with occlusions, we train the network on ShapeNet chair category and evaluate on SUN RGB-D dataset. The NOCS REAL275 dataset collects the object pose annotations of six categories, with 8K images among 18 real scenes in total. We utilize the testing split including 2750 images for the evaluation. The Wild6D dataset contains 486 videos over 162 testing objects, which are a magnitude higher than the NOCS REAL275 dataset and challenge the model generalization ability. The SUN RGB-D dataset contains 10182 9D bounding boxes for the chair category in indoor environments, including strong occluded scenes. Therefore evaluations on SUN RGB-D dataset reflect the network performance against occlusions.

4.2 Implementation Details

For the network training, only 10 synthetic models are selected for each category from ShapeNet dataset. For each synthetic object, 40 images from different views are rendered for the lifting of 3D semantic point clouds and reused for the network training. The rendered RGB-D images are of a resolution 480 x 480 and in total 400 partial rendered views are use for the synthetic training per category. The smallest DINOv2 model ViT-S [38] with a feature dimension of 384 are leveraged for the real-time inference speed. For the 2D detection masks on the evaluation dataset, we use the trained MaskRCNN [16] results provided from NOCS, Wild6D, SUN RGB-D datasets respectively. The cropped RGB-D images from 2D detections are resized to a dimension of 480 x 480 and back-projected to the semantic partial point cloud. At training and inference time, 3000 points and 1000 points are uniformly sampled from the full object model P_s and input partial point cloud Q_s for correspondence estimation. GS-Pose is trained with a learning rate of 1e-4 for 100 epochs for each category on a desktop with Intel Xeon E5-2698 CPU and Tesla V100-DGCS-32GB GPU.

4.3 Metrics

For the 9D object pose evaluation, the mean precision of 3D intersection over union (IoU) at different thresholds of X% are recorded as $3D_X$. To be noticed, methods [63], [32], [35] employ different implementations of the metric. CPPF [63] firstly finds the intersections points between two 3D bounding boxes and calculates the overlap as the convex hull volume from the intersection points, which we annotate as $3D^*_X$. Methods [32], [35] directly utilize the maximum and minimum of two bounding boxes to get the overlap, which we annotate as $3D^+_X$. Additionally 5°5cm, 10°5cm, 15°5cm metrics are reported to measure the accuracy of rotations and translations. The 20°10cm, 40°20cm, 60°30cm metrics are used for the evaluation of rotation and translation error on the SUN RGB-D dataset.

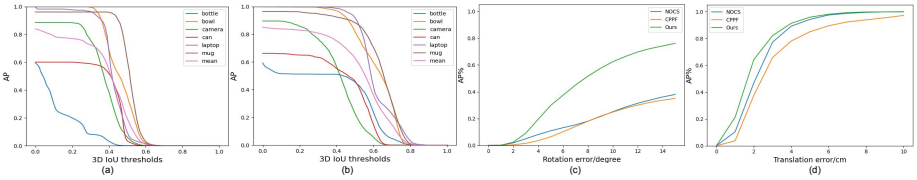


	Training Data	N(S)	$3D^*_{25}$ ↑	$3D^*_{50}$ ↑	5°5cm ↑	10°5cm ↑	15°5cm ↑
Chen et al. [5]	Syn(O)	210	15.5	1.3	0.7	3.6	9.1
Gao et al. [13]	Syn(O)	210	68.6	24.7	7.8	17.1	26.5
CPPF [63]	Syn(O)	210	78.2	26.4	16.9	44.9	50.8
CPPF++ [62]	Syn(O)	210	82.4	55.2	32.3	65.9	85.2
ZSP [15]	-	0	-	-	5.8	20.3	32.0
CPPF++ [62]	Syn(O)	**10**	80.9	51.6	22.6	59.6	**74.0**
Ours	Syn(O)	**10**	**82.1**	**63.2**	**28.8**	**60.1**	73.6

Table 1: Results for REAL275 dataset against synthetic-only and zero-shot baselines. Top: Qualitative results. The predicted 3D bounding boxes from NOCS [55] (left), CPPF [63] (middle), and ours (right). Green is predicted and red is the ground truth. Bottom: Evaluation results in comparison with synthetic-only [5,13,63] and zero-shot [15] baselines. Syn.(O) means synthetic ShapeNet objects only. N(S) represent the number of synthetic objects in the training per category.

4.4 Performance Analysis

Performance on NOCS REAL275 Dataset For thorough evaluation, we compare our method with synthetic-only approaches as well as networks trained on real annotated datasets. The comparison with synthetic-only approaches are reported in Tab. 1. The evaluation shows that our method results in an overall increase of 3D IoU, rotation and translation scores. Especially the $3D_{50}$ metric increases greatly by 36.8% in comparison with geometric-only baselines [63], even though the method is trained on a smaller amount of synthetic objects. The 5°5cm, 10°5cm, 15°5cm increase by 11.9%, 15.2%, 22.8%. It is observed that ours performs better than CPPF on difficult categories such as mugs and laptops. The detailed result for each category is plotted on the top of Tab. 2. We further compare with CPPF++ [62] which leverages both RGB and depth inputs. Evaluation results show that we outperform CPPF++ by 11.6% on $3D^*_{50}$ and 6.2% on 5°5cm when only 10 objects are available for the training. Noteworthy CPPF++ fails to fuse RGB-features and geometric features in one network, therefore they train two separate networks and get best matching results from two predictions. As a result, CPPF++ has an inference time of 930 ms per object, while ours needs only 51ms and is real-time capable. In addition we compare with ZSP [15], a semantic-only approach for zero-shot pose estimations. Ours performs much better on both the 5°5cm, 10°5cm, 15°5cm metrics.



	Training Data	Shape Prior	$3D_{75}^{+}$ ↑	5°5cm ↑	10°5cm ↑
NOCS [55]	Syn(O+B)+Real	✗	9.4	10.0	25.2
FS-Net [4]	Real	✗	-	28.2	60.8
DualPoseNet [31]	Syn(O+B)+Real	✗	30.8	35.9	66.8
GPV-Pose [8]	Real	✗	-	42.9	73.3
SS-ConvNet [29]	Syn(O+B)+Real	✗	-	43.4	63.5
VI-Net [32]	Syn(O+B)+Real	✗	**48.3**	**57.6**	**82.1**
SPD [50]	Syn(O+B)+Real	✓	27.0	21.4	54.1
CR-Net [56]	Syn(O+B)+Real	✓	33.2	34.3	60.8
CenterSnap [20]	Syn(O+B)+Real	✓	-	29.1	64.3
ACR-Pose [9]	Syn(O+B)+Real	✓	-	36.9	54.8
SGPA [3]	Syn(O+B)+Real	✓	37.1	39.6	70.7
SPD+CATRE [35]	Syn(O+B)+Real	✓	43.6	54.4	73.1
DPDN [30]	Syn(O+B)+Real	✓	-	50.7	78.4
Ours	Syn(O)	✓	37.0	28.8	60.1

Table 2: Results for REAL275 dataset against networks trained on real annotated data. Top: Visualization of each category's 3D IoUs for CPPF (a) and Ours (b). The translation and rotation mAPs of our method in comparison with CPPF and NOCS are plotted in (c) and (d). Bottom: Evaluation results in comparison with baselines trained on real annotated data. Syn.(O+B) means ShapeNet models rendered with real backgrounds (NOCS CAMERA25 dataset). Real means real images in NOCS REAL275 dataset. Syn.(O) indicates synthetic ShapeNet objects only.

The evaluation results in comparison with methods trained on real annotated datasets are reported in Tab. 2. Without the domain gap of synthetic data, VI-Net [32] trained on real data lead highest scores on the metrics. However, our method provides competitive results in comparison with methods trained on the real data, such as DualPoseNet [31], FS-Net [4], CR-Net [56], CenterSnap [20], ACR-Pose [9]. In comparison with DualPoseNet [31], the 5°5cm and 10°5cm scores are slightly lower, while ours outperforms DualPoseNet on $3D_{75}^{+}$ by 6.2%. In comparison with ACR-Pose [9], our 10°5cm is higher (60.1% vs 54.8%). The comparisons show that our method has competitive performance in comparison with approaches trained on real data, by only learning from a limited amount of synthetic models.

Performance on Wild6D Dataset Wild6D dataset [65] contains 162 testing objects, much more than NOCS dataset. The evaluation results in Tab. 4 show that our method has a strong generalization ability on novel object instances in the wild, even though trained only with a few synthetic objects. In comparison with methods trained with real data such as DualPoseNet, our method provides comparable results for the $3D_{25}$ (84.6% vs 90.0%), $3D_{50}$ (67.7% vs 70.3%), 5°5cm (30.8% vs 34.4%) metrics, and outperforms the state-of-the-art method RePoNet-semi [65] on 10°5cm by 8.5 % without using real data. Failure cases are

Metric	$3D^+_{10}$ ↑	$3D^+_{25}$ ↑	20°10cm ↑	40°20cm ↑	60°30cm ↑
CPPF [63]	36.0	14.6	1.1	7.7	13.1
Ours	**56.8**	**33.0**	**6.6**	**43.4**	**69.7**

Table 3: Results for SUN RGB-D dataset. Top: Qualitative results for predicted object poses under heavy occlusion. Green is predicted and red is the ground truth. Bottom: Evaluation results in comparison with the CPPF baseline.

Metric	Training Data	N(S)	N(R))	$3D^+_{25}$ ↑	$3D^+_{50}$ ↑	5°5cm ↑	10°5cm ↑
CASS [2]	Syn(O+B)+Real(NOCS)	180	3	19.8	1.0	0.0	0.0
SPD [50]	Syn(O+B)+Real(NOCS)	180	3	55.5	32.5	3.5	13.9
DualPoseNet [31]	Syn(O+B)+Real(NOCS)	180	3	**90.0**	70.0	22.8	36.5
RePoNet-semi [65]	Syn(O+B)+Real(Wild6D*)	180	312	84.7	**70.3**	**34.4**	42.5
Ours	**Syn(O)**	**10**	**0**	84.6	67.7	30.8	**51.0**

Table 4: Results for Wild6D dataset. Syn.(O) means synthetic ShapeNet objects only, while Syn.(O+B) means ShapeNet models rendered with real backgrounds (NOCS CAMERA25 dataset). Wild6D* means Wild6D dataset without pose annotatons. N(S) and N(R) represent the number of synthetic and real objects in the training per category.

observed when the depth estimations fail for transparent bottles, or inaccurate 2D segmentations of the cameras lead to the pose estimation failures.

Performance on SUN RGB-D Dataset SUN RGB-D dataset features challenging indoor scenes with occlusion and we evaluate on all the chairs in the validation dataset following the setup in CPPF [63]. The evaluation results are shown in Tab. 3, and our proposed method outperforms the baseline by 20.8% and 18.4% on $3D_{10}$ and $3D_{25}$ metric, which shows good generalization ability towards zero-shot object poses estimation in indoor scenes. The rotation and translation scores are higher than the baseline, especially for 40°20cm and 60°30 cm. In extremely challenging scenes where the chairs are stacked together or heavily occluded as visualized on top of Tab. 3, GS-Pose still predicts accurate results in comparison with the baseline. The experiment in SUN RGB-D dataset shows the robustness of our method in the case of occlusions.

4.5 Ablation Study

To analyse different network components, exhaustive ablations are performed and the following results are reported for the NOCS REAL275 dataset. **Inlier Probability Prediction for Matching** In the ablation A_1 from Tab. 5 (a), the inlier probability module is removed including calculation of the assignment

(a) Ablation on network components

	C1	C2	$3D^*_{25}$ ↑	$3D^*_{50}$ ↑	5°5cm ↑	10°5cm ↑	15°5cm ↑
A_1		✓	64.9	48.9	19.9	49.3	66.9
A_2	✓		72.3	44.5	12.7	40.5	56.7
A_3	✓	✓	**82.1**	**63.2**	**28.8**	**60.1**	**73.6**

(b) Ablation on the training objects number

Metric	$3D^*_{25}$ ↑	$3D^*_{50}$ ↑	5°5cm ↑	10°5cm ↑	15°5cm ↑
CPPF [63] (10 objects)	75.7	14.6	7.3	27.1	33.4
CPPF [63] (40 objects)	77.3	26.1	13.0	37.6	43.6
CPPF [63] (210 objects)	78.2	26.4	16.9	44.9	50.8
Ours (10 objects)	82.1	63.2	28.8	60.1	73.6
Ours (20 objects)	82.3	62.7	29.9	62.9	77.6
Ours (40 objects)	82.1	63.8	28.9	57.5	74.9

Table 5: Ablation study on the NOCS REAL275 dataset. (a) Ablation of network components on NOCS REAL275 dataset, C1 represents inlier probability networks and C2 stands for symmetry handling of ambiguous categories. (b) Ablation on the influence of objects number in the training.

matrix (Equ. 3) and the inlier classification loss (Equ. 4 and 5). Without consideration of matching inliers, the $3D^*_{25}$, $3D^*_{50}$ drop by 17.2% and 14.3%. In addition to the worse 3D bounding box predictions, the rotation and translation scores also decrease slightly. The 5°5cm, 10°5cm, 15°5cm decrease by 8.9%, 10.8%, 6.7%. Evaluation shows that the predicting inlier probability helps the network to focus on the regions of attention and reduces the outliers in the final matching stage. **Symmetric Handling** In the ablation A_2 from Tab. 5 (a), we remove the symmetry handling of all categories in the training. The result shows that the $3D^*_{25}$ drops slight to 72.3%, while $3D^*_{50}$ decreases greatly to 44.5%. The 5°5cm, 10°5cm, 15°5cm drop by 16.1%, 19.6%, 16.9%. The results show that the conflicting ground truth matches confuse the network and lead to inferior performances, and it is important to disambiguate the ground truth matches for the axis-symmetry categories. **Influence of Synthetic Training Object Numbers** To show the influence of the number of synthetic objects for training, we train CPPF with different object numbers and show the evaluation result in Tab. 5 (b). The 5°5cm, 10°5cm, 15°5cm score increases with the number of training objects, which shows that approaches such as CPPF that rely on geometric information and synthetic-only data require more object shape variation in the training dataset for better generalization capability. In contrast, we train our method with 10, 20, 40 synthetic models as shown in Tab. 5 (b) and the result shows that the performance saturates already with as few as 10 objects on $3D^*_{25}$ and $3D^*_{50}$. **Inference Time** The inference takes only 0.051s in average on a RTX 3080 GPU and is real-time capable, which is much faster than CPPF (0.413s) and comparable with GPV-Pose (0.05s).

5 Conclusion

In this paper, we introduce a novel 3D representation incorporating both semantic and geometric features for category-level pose estimations from synthetic objects. Based on the novel representation, we employ a matching network self-supervised from the semantic point cloud and their 2D RGB-D renderings. While requiring only ten object instances per category, our method outperforms synthetic-only baselines by a great margin and shows an outstanding generalization ability on multiple real datasets.

References

1. Chang, A.X., Funkhouser, T.A., Guibas, L.J., Hanrahan, P., Huang, Q.X., Li, Z., Savarese, S., Savva, M., Song, S., Su, H., Xiao, J., Yi, L., Yu, F.: Shapenet: An information-rich 3d model repository. ArXiv **abs/1512.03012** (2015), https://api.semanticscholar.org/CorpusID:2554264
2. Chen, D., Li, J., Xu, K.: Learning canonical shape space for category-level 6d object pose and size estimation. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) pp. 11970–11979 (2020), https://api.semanticscholar.org/CorpusID:210919925
3. Chen, K., Dou, Q.: Sgpa: Structure-guided prior adaptation for category-level 6d object pose estimation. In: IEEE/CVF International Conference on Computer Vision (ICCV). pp. 2773–2782 (2021)
4. Chen, W., Jia, X., Chang, H.J., Duan, J., Shen, L., Leonardis, A.: Fs-net: Fast shape-based network for category-level 6d object pose estimation with decoupled rotation mechanism. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1581–1590 (2021)
5. Chen, X., Dong, Z., Song, J., Geiger, A., Hilliges, O.: Category level object pose estimation via neural analysis-by-synthesis. In: European Conference on Computer Vision (ECCV). pp. 139–156. Springer (2020)
6. Deng, H., Birdal, T., Ilic, S.: Ppf-foldnet: Unsupervised learning of rotation invariant 3d local descriptors. In: European conference on computer vision (ECCV). pp. 602–618 (2018)
7. Deng, H., Birdal, T., Ilic, S.: Ppfnet: Global context aware local features for robust 3d point matching. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 195–205 (2018)
8. Di, Y., Zhang, R., Lou, Z., Manhardt, F., Ji, X., Navab, N., Tombari, F.: Gpv-pose: Category-level object pose estimation via geometry-guided point-wise voting. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR). pp. 6781–6791 (2022)
9. Fan, Z., Song, Z., Xu, J., Wang, Z., Wu, K., Liu, H., He, J.: Acr-pose: Adversarial canonical representation reconstruction network for category level 6d object pose estimation. arXiv preprint arXiv:2111.10524 (2021)
10. Fischler, M.A., Bolles, R.C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Commun. ACM **24**, 381–395 (1981), https://api.semanticscholar.org/CorpusID:972888
11. Gao, D., Li, Y., Ruhkamp, P., Skobleva, I., Wysocki, M., Jung, H., Wang, P., Guridi, A., Busam, B.: Polarimetric pose prediction. In: European Conference on Computer Vision (ECCV). pp. 735–752. Springer (2022)
12. Gao, G., Lauri, M., Wang, Y., Hu, X., Zhang, J., Frintrop, S.: 6d object pose regression via supervised learning on point clouds. In: 2020 IEEE International Conference on Robotics and Automation (ICRA). pp. 3643–3649. IEEE (2020)
13. Gao, G., Lauri, M., Wang, Y., Hu, X., Zhang, J., Frintrop, S.: 6d object pose regression via supervised learning on point clouds. International Conference on Robotics and Automation (ICRA) pp. 3643–3649 (2020), https://api.semanticscholar.org/CorpusID:210911622
14. Goodwin, W., Havoutis, I., Posner, I.: You only look at one: Category-level object representations for pose estimation from a single example. arXiv preprint arXiv:2305.12626 (2023)

15. Goodwin, W., Vaze, S., Havoutis, I., Posner, I.: Zero-shot category-level object pose estimation. In: European Conference on Computer Vision (ECCV). pp. 516–532. Springer (2022)
16. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask r-cnn. In: IEEE/CVF International Conference on Computer Vision (ICCV). pp. 2961–2969 (2017)
17. He, Y., Huang, H., Fan, H., Chen, Q., Sun, J.: Ffb6d: A full flow bidirectional fusion network for 6d pose estimation. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 3003–3013 (2021)
18. He, Y., Sun, W., Huang, H., Liu, J., Fan, H., Sun, J.: Pvn3d: A deep point-wise 3d keypoints voting network for 6dof pose estimation. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 11632–11641 (2020)
19. Hodan, T., Barath, D., Matas, J.: EPOS: Estimating 6D pose of objects with symmetries. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 11703–11712. IEEE (Jun 2020)
20. Irshad, M.Z., Kollar, T., Laskey, M., Stone, K., Kira, Z.: Centersnap: Single-shot multi-object 3d shape reconstruction and categorical 6d pose and size estimation. In: International Conference on Robotics and Automation (ICRA). pp. 10632–10640. IEEE (2022)
21. Karnati, M., Seal, A., Yazidi, A., Krejcar, O.: Lienet: A deep convolution neural network framework for detecting deception. IEEE Transactions on Cognitive and Developmental Systems **14**(3), 971–984 (2021)
22. Kehl, W., Manhardt, F., Tombari, F., Ilic, S., Navab, N.: Ssd-6d: Making rgb-based 3d detection and 6d pose estimation great again. IEEE International Conference on Computer Vision (ICCV) pp. 1530–1538 (2017), `https://api.semanticscholar.org/CorpusID:10655945`
23. Labbé, Y., Carpentier, J., Aubry, M., Sivic, J.: Cosypose: Consistent multi-view multi-object 6d pose estimation. In: European Conference on Computer Vision (ECCV). pp. 574–591. Springer (2020)
24. Lee, T., Lee, B.U., Shin, I., Choe, J., Shin, U., Kweon, I.S., Yoon, K.J.: Udacope: unsupervised domain adaptation for category-level object pose estimation. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 14891–14900 (2022)
25. Lee, T., Tremblay, J., Blukis, V., Wen, B., Lee, B.U., Shin, I., Birchfield, S., Kweon, I.S., Yoon, K.J.: Tta-cope: Test-time adaptation for category-level object pose estimation. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 21285–21295 (2023)
26. Li, X., Weng, Y., Yi, L., Guibas, L.J., Abbott, A., Song, S., Wang, H.: Leveraging se (3) equivariance for self-supervised category-level object pose estimation from point clouds. Advances in neural information processing systems **34**, 15370–15381 (2021)
27. Li, Y., Wang, G., Ji, X., Xiang, Y., Fox, D.: Deepim: Deep iterative matching for 6d pose estimation. In: European Conference on Computer Vision (ECCV). pp. 683–698 (2018)
28. Li, Z., Wang, G., Ji, X.: Cdpn: Coordinates-based disentangled pose network for real-time rgb-based 6-dof object pose estimation. In: IEEE/CVF International Conference on Computer Vision (ICCV). pp. 7678–7687 (2019)
29. Lin, J., Li, H., Chen, K., Lu, J., Jia, K.: Sparse steerable convolutions: An efficient learning of se (3)-equivariant features for estimation and tracking of object poses in 3d space. Advances in Neural Information Processing Systems **34**, 16779–16790 (2021)

30. Lin, J., Wei, Z., Ding, C., Jia, K.: Category-level 6d object pose and size estimation using self-supervised deep prior deformation networks. In: European Conference on Computer Vision (ECCV). pp. 19–34. Springer (2022)
31. Lin, J., Wei, Z., Li, Z., Xu, S., Jia, K., Li, Y.: Dualposenet: Category-level 6d object pose and size estimation using dual pose network with refined learning of pose consistency. In: IEEE/CVF International Conference on Computer Vision (CVPR). pp. 3560–3569 (2021)
32. Lin, J., Wei, Z., Zhang, Y., Jia, K.: Vi-net: Boosting category-level 6d object pose estimation via learning decoupled rotations on the spherical representations. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 14001–14011 (2023)
33. Lin, T.Y., Goyal, P., Girshick, R.B., He, K., Dollár, P.: Focal loss for dense object detection. IEEE/CVF International Conference on Computer Vision (ICCV) pp. 2999–3007 (2017), https://api.semanticscholar.org/CorpusID:47252984
34. Lindenberger, P., Sarlin, P.E., Pollefeys, M.: Lightglue: Local feature matching at light speed. arXiv preprint arXiv:2306.13643 (2023)
35. Liu, X., Wang, G., Li, Y., Ji, X.: Catre: Iterative point clouds alignment for category-level object pose refinement. In: European Conference on Computer Vision (ECCV). pp. 499–516. Springer (2022)
36. Manhardt, F., Wang, G., Busam, B., Nickel, M., Meier, S., Minciullo, L., Ji, X., Navab, N.: Cps++: Improving class-level 6d pose and shape estimation from monocular images with self-supervised learning. arXiv preprint arXiv:2003.05848 (2020)
37. Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., et al.: Language models are few-shot learners. arXiv preprint arXiv:2005.14165 (2020)
38. Oquab, M., Darcet, T., Moutakanni, T., Vo, H.Q., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., Assran, M., Ballas, N., Galuba, W., Howes, R., Huang, P.Y.B., Li, S.W., Misra, I., Rabbat, M.G., Sharma, V., Synnaeve, G., Xu, H., Jégou, H., Mairal, J., Labatut, P., Joulin, A., Bojanowski, P.: Dinov2: Learning robust visual features without supervision. ArXiv **abs/2304.07193** (2023), https://api.semanticscholar.org/CorpusID:258170077
39. Pan, P., Fan, Z., Feng, B.Y., Wang, P., Li, C., Wang, Z.: Learning to estimate 6dof pose from limited data: A few-shot, generalizable approach using rgb images. arXiv preprint arXiv:2306.07598 (2023)
40. Park, K., Patten, T., Vincze, M.: Pix2pose: Pixel-wise coordinate regression of objects for 6d pose estimation. In: IEEE/CVF International Conference on Computer Vision (ICCV). pp. 7668–7677 (2019)
41. Peng, S., Liu, Y., Huang, Q., Zhou, X., Bao, H.: Pvnet: Pixel-wise voting network for 6dof pose estimation. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 4561–4570 (2019)
42. Qi, C.R., Su, H., Mo, K., Guibas, L.J.: Pointnet: Deep learning on point sets for 3d classification and segmentation. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 652–660 (2017)
43. Qin, Z., Yu, H., Wang, C., Guo, Y., Peng, Y., Xu, K.: Geometric transformer for fast and robust point cloud registration. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 11143–11152 (2022)
44. Rad, M., Lepetit, V.: Bb8: A scalable, accurate, robust to partial occlusion method for predicting the 3d poses of challenging objects without using depth. In: Proceedings of the IEEE international conference on computer vision. pp. 3828–3836 (2017)

45. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International conference on machine learning. pp. 8748–8763. PMLR (2021)
46. Sarlin, P.E., DeTone, D., Malisiewicz, T., Rabinovich, A.: Superglue: Learning feature matching with graph neural networks. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 4938–4947 (2020)
47. Song, C., Song, J., Huang, Q.: Hybridpose: 6d object pose estimation under hybrid representations. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 431–440 (2020)
48. Song, S., Lichtenberg, S.P., Xiao, J.: Sun rgb-d: A rgb-d scene understanding benchmark suite. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) pp. 567–576 (2015), https://api.semanticscholar.org/CorpusID:6242669
49. Sundermeyer, M., Marton, Z.C., Durner, M., Brucker, M., Triebel, R.: Implicit 3d orientation learning for 6d object detection from rgb images. In: European Conference on Computer Vision (ECCV). pp. 699–715 (2018)
50. Tian, M., Ang Jr, M.H., Lee, G.H.: Shape prior deformation for categorical 6d object pose and size estimation. In: European Conference on Computer Vision (ECCV) (August 2020)
51. Umeyama, S.: Least-squares estimation of transformation parameters between two point patterns. IEEE Trans. Pattern Anal. Mach. Intell. **13**, 376–380 (1991), https://api.semanticscholar.org/CorpusID:206421766
52. Wang, C., Xu, D., Zhu, Y., Martín-Martín, R., Lu, C., Fei-Fei, L., Savarese, S.: Densefusion: 6d object pose estimation by iterative dense fusion. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 3343–3352 (2019)
53. Wang, G., Manhardt, F., Shao, J., Ji, X., Navab, N., Tombari, F.: Self6d: Self-supervised monocular 6d object pose estimation. European Conference on Computer Vision (ECCV) **abs/2004.06468** (2020), https://api.semanticscholar.org/CorpusID:215754192
54. Wang, G., Manhardt, F., Tombari, F., Ji, X.: Gdr-net: Geometry-guided direct regression network for monocular 6d object pose estimation. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 16611–16621 (2021)
55. Wang, H., Sridhar, S., Huang, J., Valentin, J.P.C., Song, S., Guibas, L.J.: Normalized object coordinate space for category-level 6d object pose and size estimation. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) pp. 2637–2646 (2019), https://api.semanticscholar.org/CorpusID:57761160
56. Wang, J., Chen, K., Dou, Q.: Category-level 6d object pose estimation via cascaded relation and recurrent reconstruction networks. In: 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 4807–4814. IEEE (2021)
57. Wang, P., Garattoni, L., Meier, S., Navab, N., Busam, B.: Crocps: Addressing photometric challenges in self-supervised category-level 6d object poses with cross-modal learning. In: British Machine Vision Conference (2022), https://api.semanticscholar.org/CorpusID:256903232
58. Wang, P., Jung, H., Li, Y., Shen, S., Srikanth, R.P., Garattoni, L., Meier, S., Navab, N., Busam, B.: Phocal: A multi-modal dataset for category-level object pose estimation with photometrically challenging objects. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 21222–21231 (2022)

59. Wang, P., Manhardt, F., Minciullo, L., Garattoni, L., Meier, S., Navab, N., Busam, B.: Demograsp: Few-shot learning for robotic grasping with human demonstration. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 5733–5740. IEEE (2021)
60. Xiang, Y., Schmidt, T., Narayanan, V., Fox, D.: Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes. arXiv preprint arXiv:1711.00199 (2017)
61. Xu, Y., Lin, K.Y., Zhang, G., Wang, X., Li, H.: RNNPose: Recurrent 6-DoF object pose refinement with robust correspondence field estimation and pose optimization. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 14880–14890. IEEE (Jun 2022)
62. You, Y., He, W., Liu, J., Xiong, H., Wang, W., Lu, C.: Cppf++: Uncertainty-aware sim2real object pose estimation by vote aggregation. IEEE Transactions on Pattern Analysis and Machine Intelligence (2024)
63. You, Y., Shi, R., Wang, W., Lu, C.: Cppf: Towards robust category-level 9d pose estimation in the wild. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) pp. 6856–6865 (2022), https://api.semanticscholar.org/CorpusID:247291938
64. Zakharov, S., Shugurov, I., Ilic, S.: Dpod: 6d pose object detector and refiner. In: IEEE/CVF International Conference on Computer Vision (ICCV). pp. 1941–1950 (2019)
65. Ze, Y., Wang, X.: Category-level 6d object pose estimation in the wild: A semi-supervised learning approach and a new dataset. Advances in Neural Information Processing Systems **35**, 27469–27483 (2022)
66. Zhang, R., Di, Y., Lou, Z., Manhardt, F., Navab, N., Tombari, F., Ji, X.: Rbp-pose: Residual bounding box projection for category-level pose estimation. ArXiv **abs/2208.00237** (2022), https://api.semanticscholar.org/CorpusID:251223949
67. Zhao, C., Hu, Y., Salzmann, M.: Fusing local similarities for retrieval-based 3d orientation estimation of unseen objects. ArXiv **abs/2203.08472** (2022), https://api.semanticscholar.org/CorpusID:247475898