SeqPose: An End-to-End Framework to Unify Single-frame and Video-based RGB Category-Level Pose Estimation

Yuzhu Ji[1,*], **Mingshan Sun**[2,*], **Jianyang Shi**[3], **Xiaoke Jiang**[4], **Yiqun Zhang**[1,5†], **Haijun Zhang**[6]
[1]Guangdong University of Technology, [2]CVTE Research, [3]Harbin Institute of Technology
[4]International Digital Economy Academy (IDEA), [5]Hong Kong Baptist University
yuzhu.ji@gdut.edu.cn, mingshine.sun@gmail.com, jianyangshi02@gmail.com,
jiangxiaoke@idea.edu.cn, yqzhang@gdut.edu.cn, hjzhang@hit.edu.cn

Abstract

Category-level object pose estimation is a long-standing and fundamental task crucial for augmented reality and robotic manipulation applications. Existing RGB-based approaches struggle with multi-stage settings and heavily rely on off-the-shelf techniques, such as object detectors, depth estimators, non-differentiable NOCS shape alignment, etc. Extra dependencies lead to the accumulation of errors and complicate the whole pipeline, limiting the deployment of these approaches in practical applications. This paper streamlined an end-to-end framework unifying the single-frame and video-based category-level pose estimation. Specifically, instead of explicitly introducing extra dependencies, the DINOv2 encoder and depth decoder, as robust semantic and geometric prior extractors, are leveraged to produce intra-frame hierarchical semantic and geometric features. A spatial-temporal sparse query network is developed to model the implicit correspondence and inter-frame correlations between a set of implicit 3D query anchors and intra-frame features. Finally, a pose prediction head is employed using the bi-partite matching algorithm. Experimental results demonstrate that our model achieves state-of-the-art performance compared with RGB-based categorical pose estimation methods on the REAL275 and CAMERA25 datasets. Our code is available at https://andrewchiyz.github.io/vision.3dv.seqpose/.

1 Introduction

Category-level object pose estimation is a fundamental computer vision task and crucial for 3D perception and understanding [Liu *et al.*, 2024]. It aims to accurately estimate the locations, orientations, and metric sizes of the objects w.r.t. the camera in given images, depth maps and point clouds [Zheng *et al.*, 2024; Sun *et al.*, 2024]. In practice, automatic estimation of the rotation and translation of an object can facilitate applications, such as interactions in augmented reality [Su *et al.*, 2019; Wen *et al.*, 2023], robotic manipulation [Liu *et al.*, 2023a; Yu *et al.*, 2023; Liu *et al.*, 2023b; Sun *et al.*, 2022] and autonomous driving [Hoque *et al.*, 2023; Sun *et al.*, 2023]. Recent advances in RGBD-based categorical object pose estimation have achieved remarkable progress [Wang *et al.*, 2019; Liu *et al.*, 2023c; Chen *et al.*, 2024; Lin *et al.*, 2024; Wang *et al.*, 2024]. However, RGBD-based methods largely depend on depth sensors [Zhang *et al.*, 2024], which limits their applications in practice, especially for RGB-only settings without depth devices [Liu *et al.*, 2024]. Recently, RGB-based categorical pose estimation has also demonstrated its wide application scenarios equipped with mobile devices, such as AR headsets, smartphones, *etc*., and become a promising trend [Liu *et al.*, 2024].

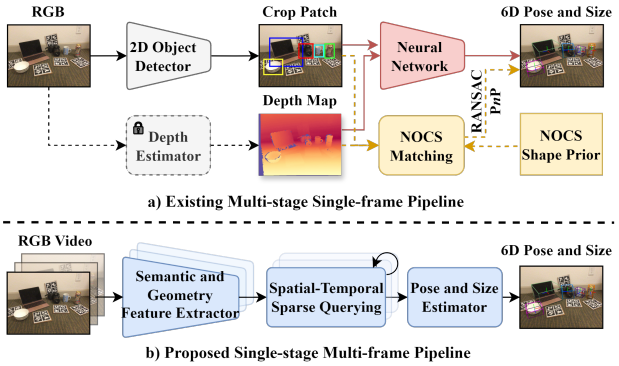


Figure 1: Comparison with existing pipelines: a) Multi-stage single-frame pipeline for RGB-based category-level pose and size estimation. The Red arrow indicates the two-stage direct regression approaches and the yellow arrow with dash lines refers to the multi-stage NOCS-based pipelines. b) Our streamlined compact single-stage multi-frame pipeline.

RGB-based methods [Manhardt *et al.*, 2020; Lee *et al.*, 2021; Fan *et al.*, 2022; Wei *et al.*, 2024] have been proposed in recent years. Concretely, the RGB-based category-level pose estimation methods can be roughly classified into two main paradigms: 1) two-stage direct regression methods, and 2) multi-stage normalized object coordinate space (NOCS) based methods (see Figure 1 (a)). Particularly, two-stage direct regression methods achieved pose estimation by leveraging multiple separate models, such as object detector and

[*]Co-first author
[†]Corresponding author

pose regressor. Multi-stage NOCS-based methods estimate the shape variations by capturing the correspondence between the observed objects and NOCS representation. Object pose is calculated using the Umeyama [Umeyama, 1991] and Perspective-n-Point (P*n*P) [Lepetit *et al.*, 2009] algorithms.

However, existing RGB-based approaches struggle with multi-stage settings and heavily rely on intermediate results of single-frame object detection and depth estimation. Furthermore, the dependency on off-the-shelf techniques, including 2D object detectors, depth estimators, non-differentiable NOCS shape alignment, *etc*., leads to cumulative errors, degrading the performance of accuracy and stability of the pose and size estimation. Additionally, the multi-stage settings and extra dependencies complicate the whole pipeline and impose significant limitations on their practical applications and deployment in real-world scenarios. It thus motivates us to streamline a compact yet efficient end-to-end framework for category-level 6D pose and size estimation.

To achieve this goal, we propose SeqPose, a Spatial-tEmporal sparse Query network for RGB-based category-level object Pose estimation. Our *key insight* is that: 1) Despite the limited information and ill-posed problem setting, advanced feature extractors can be used to effectively map RGB inputs to implicit robust semantic and geometry feature space without explicitly introducing object detectors or depth estimators; 2) The idea of using the spatial-temporal correlations for RGB-based categorical object pose estimation remains underexplored. It is promising to introduce spatial-temporal information as a strong prior, to correct geometry errors and stabilize the pose and size estimation results; 3) The intra-frame semantic and geometry feature can be aggregated by leveraging a set of sparse 3D query anchors and propagating inter-frame correspondence, which thereby unifies single-frame and video-based RGB category-level object pose estimation. Our SeqPose comprises three main modules: (a) A hierarchical multi-scale feature fusion module that leverages the DINOv2 encoder and a depth decoder to integrate robust intra-frame semantic and geometric features. (b) A spatial-temporal sparse query network for aggregating intra-frame features by a set of inter-frame correlated 3D query anchors. (c) A pose prediction head to directly estimate pose and size from the learned queries. To this end, our SeqPose can be trained end-to-end and enables shape-prior-free categorical pose estimation. Experimental results show that our model can achieve superior performance in estimating 6D pose and size for category-level objects on both single-frame and video-based RGB benchmark datasets. Our main contributions can be summarized as follows:

- To the best of our knowledge, we are the first to unify single-frame and video-based RGB category-level object pose and size estimation into a single-stage end-to-end framework.
- We build a DINOv2-based encoder-decoder network to learn implicit semantic categorical prior and instance-specific geometric representations by integrating hierarchical intra-frame features.
- A spatial-temporal sparse query-based network is presented to aggregate intra-frame semantic and geometry features and propagate inter-frame correlations by leveraging implicit 3D query anchors.

2 Related Work

RGBD-based Method. Initially, Wang *et al*. [Wang *et al.*, 2019] introduced the NOCS to align all objects with the same category in a shared 3D canonical representation. Specifically, in [Wang *et al.*, 2019], Mask R-CNN [He *et al.*, 2017] is adopted to segment objects and an extra head is designed to predict the 2D projected NOCS map. Then, the given depth map is used to capture the 3D-3D correspondence by back-projecting the predicted NOCS map. However, the intra-class shape variations significantly impact the performance of 6D pose and size estimation for shared shape prior-based methods. Therefore, Chen *et al*. [Chen *et al.*, 2020a] proposed to learn an implicitly canonical shape space (CASS) using a variational auto-encoder. Similarly, Tian *et al*. [Tian *et al.*, 2020] proposed to reconstruct the observed 3D object to model the deformation. Furthermore, Wang *et al*. [Wang *et al.*, 2021a] presented a cascaded relation and recurrent reconstruction network to accurately recover the instance 3D model in the shared canonical space, Chen and Dou [Chen and Dou, 2021] developed a structure-guided prior adaptation scheme to adapt the 3D canonical prior model.

The major challenges for category-level pose estimation are the intra-class shape and texture variations. Matching the instance-specific shape to the mean shape prior efficiently and accurately becomes the focus for RGBD-based approaches. Moreover, the shared canonical space, *i.e*., NOCS for each category should be obtained in advance, and instance-specific geometry representation , such as patch-wise point cloud and depth map, should be calculated online or acquired offline.

RGB-based Method. In practice, depth information may be absent. It is crucial for achieving high-precision category-level pose estimation using RGB images in real applications [Liu *et al.*, 2024; Chen *et al.*, 2020b; Manhardt *et al.*, 2020; Fan *et al.*, 2022; Wei *et al.*, 2024]. Specifically, Chen *et al*. [Chen *et al.*, 2020b] proposed a VAE-based model to reconstruct the RGB image conditional on the pose. Manhardt *et al*. [Manhardt *et al.*, 2020] presented CPS++, a self-supervised model for class-level pose and metric size estimation. Lee *et al*. [Lee *et al.*, 2021] proposed a two-branch network to simultaneously regress the metric scale and NOCS map. Similarly, Fan *et al*. [Fan *et al.*, 2022] developed OLD-Net to predict the depth and NOCS representation from an RGB image for pose estimation. Wei *et al*. [Wei *et al.*, 2024] decoupled the 6D pose and size estimation by designing a 2D-3D correspondence learning and a metric scale recovery module. Zhang *et al*. [Zhang *et al.*, 2024] proposed the Laplacian mixture model to represent the object shape for pose estimation. Li *et al*. [Li *et al.*, 2024] presented a coarse-to-fine method that leveraged geometry supervision for coarse 3D feature extraction and refined the feature using pose and size constraints.

Generally, RGB-based methods depend heavily on object detection, depth estimation, and non-differentiable shared-shape-prior matching and optimization, which separates the pipeline into multiple stages and hinders these approaches

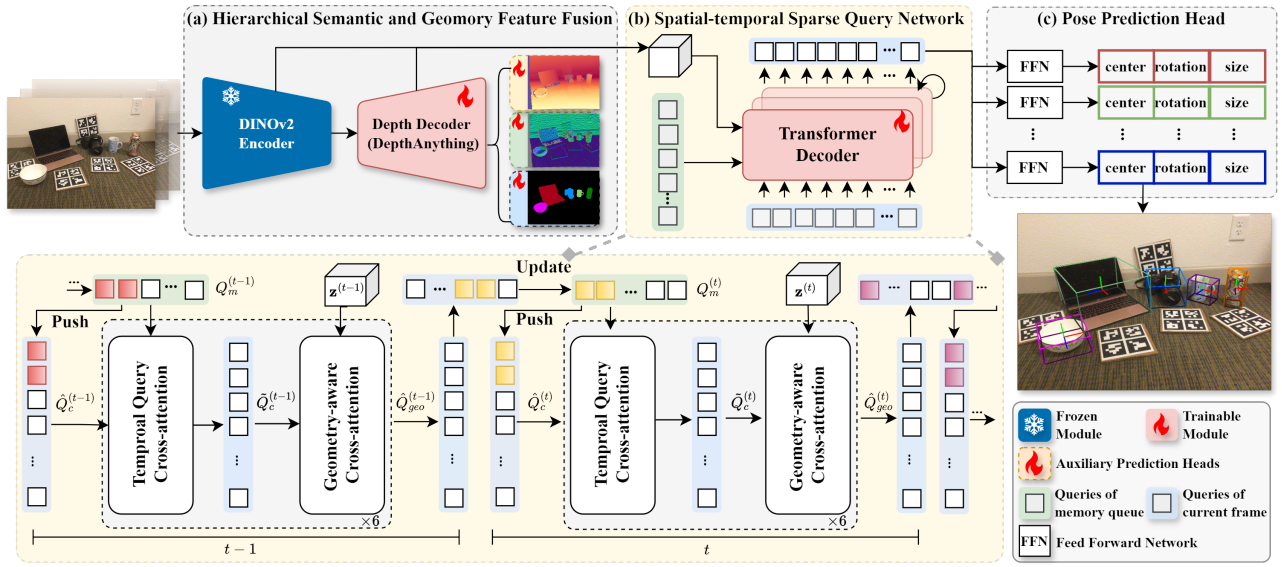


Figure 2: Overview of our proposed framework. Our framework integrates the DINOv2 encoder and a depth decoder to extract hierarchical semantic and geometry-aware features (a). A spatial-temporal sparse query network (b) aggregates intra-frame features using inter-frame correlations of implicit 3D query anchors. The pose prediction head (c) directly regresses the object pose and size from the learned queries.

from being integrated into an end-to-end framework imposing limitations in practical application and deployment.

Large Vision Models for Pose Estimation. Recently, large vision models (LVMs) have achieved remarkable progress, driven by rapid advancements in self-supervised learning techniques and the availability of large-scale datasets. DINO [Caron *et al.*, 2021; Oquab *et al.*, 2023], as a representative LVM, has gained considerable attention due to its extraordinary performance in producing semantically consistent patch-wise features. The DINO feature has been explored for detection [Liu *et al.*, 2023d], depth estimation [Yang *et al.*, 2024a; Yang *et al.*, 2024b], pose estimation [Chen *et al.*, 2024], etc. Specifically, Chen *et al.* [Chen *et al.*, 2024] proposed to learn categorical SE(3)-consistent features by integrating semantic category priors extracted using DINOv2, with object-specific geometric features. Our implemented model is closely related to [Chen *et al.*, 2024]. However, instead of introducing categorical shape priors, we propose to learn the intra-frame hierarchical representations of the semantic category priors and instance-specific geometric features produced by the DINOv2 encoder [Oquab *et al.*, 2023] and DepthAnythingV2 decoder [Yang *et al.*, 2024b], respectively. Moreover, a spatial-temporal sparse query network is designed to learn implicit 3D anchors and estimate the 6D pose and size by aggregating the intra-frame features with spatial-temporal correlated queries.

3 Method

3.1 Model Overview

Given an input sequence of monocular RGB images $\left[\mathbf{I}^{(1)}, \mathbf{I}^{(2)}, ..., \mathbf{I}^{(t)}, ..., \mathbf{I}^{(T)}\right]$, our goal is to estimate the 3D location, orientation and size of objects in each image directly. The overview of the proposed SeqPose is illustrated in Figure 2. It mainly consists of three modules: (a) A hierarchical feature fusion module is presented to learn and integrate robust *semantic* and *geometry* features using DINOv2 and the depth decoder of DepthAnythingV2, respectively. (b) A spatial-temporal sparse query network is designed to aggregate the semantic and geometry features by learning a set of inter-frame correlated 3D query anchors. Finally, (c) A 9-DoF prediction head is introduced to estimate the pose and size directly from the learned queries using the bipartite matching algorithm [Wang *et al.*, 2021b].

3.2 Hierarchical Semantic and Geometry Feature Fusion

Instead of using a CNN-based object detector, we adopt the encoder of DINOv2 [Oquab *et al.*, 2023] and decoder of DepthAnythingv2 [Yang *et al.*, 2024b] to produce intra-frame latent *semantic* and *geometry* feature. Our *key assumption* is that features produced by the DINOv2 encoder can be regarded as robust semantic priors to understand objects. Moreover, DepthAnythingv2 also demonstrates its superiority in recovering geometry by estimating accurate depth maps. In particular, the encoder of DINOv2 is responsible for extracting semantic features of an input image:

$$\mathbf{S}^{(t)} = \left[\mathbf{s}_0^{(t)}, \mathbf{s}_1^{(t)}, \cdots, \mathbf{s}_l^{(t)}\right] = \texttt{DINOv2}_{\texttt{enc}}(\mathbf{I}^{(t)}), \quad (1)$$

where $\mathbf{I}^{(t)}$ is the t-th frame in a given RGB sequence. $\mathbf{s}_i^{(t)}$ denotes i-th level of features produced by the encoder of the DINOv2 ($\texttt{DINOv2}_{\texttt{enc}}(\cdot)$). Then, the decoder of DepthAnythingv2 with convolutional layers ($\texttt{Depth}_{\texttt{dec}}(\cdot)$) is exploited for iteratively upsampling the hierarchical geometry-aware depth features, which can be expressed as:

$$\mathbf{g}_i^{(t)} = \begin{cases} \texttt{FUSE}_i(\texttt{ADD}(\mathbf{g}_{i+1}^{(t)}, \mathbf{s}_i^{(t)})), & i = l-1, \cdots, 1, 0 \\ \texttt{FUSE}_i(\mathbf{s}_i^{(t)}), & i = l \end{cases}, \quad (2)$$

where $\mathbf{g}_i^{(t)}$ refers i-th level of geometry features produced by the $\texttt{Depth}_{\texttt{dec}}(\cdot)$, which is a hierarchical feature pyramid structure with skip connection $\texttt{ADD}(\cdot)$ and residual convolutional blocks $\texttt{FUSE}_i(\cdot)$. We ignored the upsampling and reshaping operations to simplify the Equation. Finally, the semantic and geometry features are integrated by:

$$\mathbf{z}_i^{(t)} = \texttt{CONV}_i(\texttt{CAT}(\mathbf{s}_i^{(t)}, \mathbf{g}_i^{(t)})), \tag{3}$$

where $\mathbf{z}_i^{(t)}$ refers to a type of 2.5D features that integrate the i-th level of intra-frame semantic and geometry features. $\texttt{CONV}_i$, $\texttt{CAT}$ denote 1-by-1 convolutional layer and concatenation operation, respectively.

To learn semantic and geometry features, three separate pixel-wise dense prediction heads are introduced, including a depth head, a normal head and a semantic head to predict depth map $\hat{\mathbf{P}}_{\text{depth}}^{(t)}$, normal map $\hat{\mathbf{P}}_{\text{norm}}^{(t)}$, and semantic mask $\hat{\mathbf{P}}_{\text{seg}}^{(t)}$, respectively:

$$[\hat{\mathbf{P}}_{\text{seg}}^{(t)}, \hat{\mathbf{P}}_{\text{depth}}^{(t)}, \hat{\mathbf{P}}_{\text{norm}}^{(t)}] = \mathcal{P}(\mathbf{z}_0^{(t)}, \mathbf{g}_0^{(t)}). \tag{4}$$

3.3 Spatial-temporal Sparse Query Network

Our proposed framework's core is a spatial-temporal sparse query (STSQ) network, which learns to capture the implicit correspondence between a set of learnable 3D query anchors and the integrated intra-frame 2.5D features. Inspired by the query-based models in object detection [Carion *et al.*, 2020; Wang *et al.*, 2021b; Wang *et al.*, 2023], our proposed STSQ, as illustrated in Figure 2, consists of three main components: 1) A sparse query-based memory queue to learn a set of spatial-temporal correlated 3D query anchors. 2) A temporal query cross-attention module for modeling inter-frame interactions between 3D query anchors across different time steps. 3) A geometry-aware cross-attention block is introduced for aggregating intra-frame hierarchical 2.5D features guided by the estimated depth and normal map.

(a) Sparse Query-based Memory Queue. We initialize a set of implicit 3D query anchors Q_m to all zeros at time step 0 as the sparse query-based memory queue. The number of queries in the memory queue is set to $N \times K$, where N denotes the number of historical frames, K denotes the top-K candidate query anchors stored in the memory queue for each frame of a sequence. During training, the memory queue Q_m is updated according to 1) input frames within the sequence $[t - \Delta t, t - 1]$, and 2) the top-K queries within each frame ranked by the confidence score predicted by the pose prediction head. This can be formulated as:

$$Q_m^{(t)} = \texttt{CAT}(\texttt{TOP}_\texttt{K}(Q_c^{[t-\Delta t, t-1]})), t > \Delta t \tag{5}$$

where $\texttt{TOP}_\texttt{K}(\cdot)$ denotes the selection of top-K candidate queries for each frame in the sequence $[t - \Delta t, t - 1]$. $Q_c^{(t)}$ is the implicit 3D query anchors for current frame. Furthermore, the top-K object queries of the previous frame in the memory query will be selected and concatenated with the 3D query anchors $Q_c^{(t)}$ to ensure high recall:

$$\hat{Q}_c^{(t)} = \texttt{CAT}(\texttt{TOP}_\texttt{K}(Q_c^{(t-1)}), Q_c^{(t)}). \tag{6}$$

(b) Temporal Query Cross-attention. Once we get the sparse query $Q_m^{(t)}$ in the memory queue and the implicit 3D query anchors $\hat{Q}_c^{(t)}$, the temporal queries will be propagated to the current frame by using a cross-attention block:

$$\tilde{Q}_c^{(t)} = \texttt{CA}(\psi_{query}(\hat{Q}_c^{(t)}), \psi_{key}(\hat{Q}_m^{(t)}), \psi_{value}(\hat{Q}_m^{(t)})), \tag{7}$$

where $\texttt{CA}(\cdot)$ denotes the multi-head cross-attention layer. $\psi_{query}(\cdot)$, $\psi_{key}(\cdot)$ and $\psi_{value}(\cdot)$ refer the linear projections for *query*, *key*, and *value* input, respectively. $\hat{Q}_m^{(t)} = \texttt{CAT}(Q_c^{(t)}, Q_m^{(t)})$ are treated as the *key*, and *value* for introducing temporal interactions. The temporal correlations between the memory queue and the current query set are captured in the attention matrix generated by using a `softmax` function. Before feeding the sparse queries into the cross-attention layer, a pose-aware layer normalization operation is adopted for implicit camera pose information compensation. Notably, when the model operates under the single-frame (SF) setting, the historical memory queue is disabled. Therefore, $\hat{Q}_m^{(t)} = \hat{Q}_c^{(t)} = Q_c^{(t)}$. We refer the reader to the *supplementary document* for more details.

(c) Geometry-aware Cross-attention. To aggregate the spatial-temporal feature, we present a geometry-aware cross-attention block to query intra-frame 2.5D features $\mathbf{z}^{(t)}$. For simplicity, we ignore the subscript i in the following descriptions. The geometry-aware features are regarded as *key* and *value* and fed into the multi-head cross-attention block to aggregate intra-frame long-range contextual features. Considering computational and memory efficiency, a deformable cross-attention block ($\texttt{DeformCA}(\cdot)$) [Zhu *et al.*, 2021] is adopted to integrate multi-scale intra-frame features by learning a small set of sampling locations:

$$\hat{Q}_{geo}^{(t)} = \texttt{DeformCA}(\phi_{query}(\tilde{Q}_c^{(t)}), \phi_{key}(\mathbf{z}^{(t)}), \phi_{value}(\mathbf{z}^{(t)})), \tag{8}$$

where $\phi_{query}(\cdot)$, $\phi_{key}(\cdot)$ and $\phi_{value}(\cdot)$ are the linear projections for *query*, *key*, and *value*, respectively. In our implementation, the $\texttt{DeformCA}(\cdot)$ first learns to sample a small set of 3D locations in the camera coordinate system for the 3D query anchors, and then projects the 3D coordinates to the 2D image plane using the camera intrinsic.

3.4 Pose Prediction Head and Loss Function

We adopted a set prediction head to estimate 3D translation, 3D rotation and object metric size. Similar to [Carion *et al.*, 2020; Wang *et al.*, 2021b], the Hungarian algorithm is utilized to find the optimal bipartite matching between the predicted object queries and the ground-truth (GT) object sets. Concretely, a feed-forward network (FFN) is introduced to regress the 3D translation, 3D rotation, and metric size. A linear projection layer predicts the confidence score of the corresponding class label:

$$[(\hat{\mu}, \hat{\gamma}, \hat{\sigma}), \hat{y}]^{(t)} = [\texttt{FFN}(\hat{Q}_{geo}^{(t)}), \varphi(\hat{Q}_{geo}^{(t)})], \tag{9}$$

where $\hat{\mu} \in \mathbb{R}^3$, $\hat{\gamma} \in \mathbb{R}^6$, and $\hat{\sigma} \in \mathbb{R}^3$ denote the 3D translation, rotation and metric size concerning width, height and length of object queries. $\varphi(\cdot)$ is the linear projection to predict the class label $\hat{y}$. Following [Zhou *et al.*, 2019], we predict 6D continuous parameters for estimating 3D rotations rather than 4D quaternion values in our implementation.

To train our model, multiple loss functions are used, including 1) The Hungarian matching loss for optimizing the

	REAL275 (Video)								CAMERA25 (Single-frame)							
	IoU25	IoU50	IoU75	10cm	10°	10°10cm	10°5cm	5°5cm	IoU25	IoU50	IoU75	10cm	10°	10°10cm	10°5cm	5°5cm
Synthesis (ECCV'20)	-	-	-	34	14.2	4.8	-	-	-	-	-	-	-	-	-	-
CPS++ (Arxiv'20)	54.3	17.7	-	-	-	22.3	-	-	26.7	8.1	-	-	-	27.4	-	-
MSOS (RAL'21)	62	23.4	3	39.5	29.2	9.6	-	-	75.5	32.4	5.1	29.7	60.8	19.2	-	-
OLD-Net (ECCV'22)	68.7	25.4	1.9	38.9	37	9.8	-	-	74.3	32.1	5.4	30.1	74	23.4	-	-
FAP-Net (ICRA'24)	**74.2**	36.8	5.2	49.7	49.6	24.5	-	-	**81.4**	39.2	6.7	36	80.4	29.8	-	-
LaPose (ECCV'24)	65.3	31.5	8.4	42.4	**72.3**	29	11.5	5.6	32.3	12.8	2.3	12.3	72.1	11.2	3.1	2.4
DMSR (ICRA'24)	52	28.3	6.1	37.3	59.5	23.6	9.9	5.8	75.2	34.6	6.5	32.3	81.4	27.4	8.1	5.7
Ours (SF)	63	34.1	6.1	60.7	60.3	39.2	16.2	8.1	68.4	**54.4**	**12.6**	**68.2**	**82.3**	**54.5**	**11.4**	**10.6**
Ours (MF)	68.1	**41.1**	**8.5**	**63.3**	64.9	**41.8**	**17.2**	**10.4**	-	-	-	-	-	-	-	-

Table 1: Quantitative comparison with SOTA RGB-based methods on the REAL275 and CAMERA25 datasets.

bipartite matching, in which we use the focal loss $\mathcal{L}_{\text{class}}$ for supervising the category prediction, and the L1 loss for pose and size estimation. 2) The auxiliary smooth L1 losses $\mathcal{L}_{\text{depth}}$ and $\mathcal{L}_{\text{normal}}$ for depth and normal map estimation. 3) The cross-entropy (CE) loss $\mathcal{L}_{\text{seg}}$ for semantic segmentation. The overall loss can be defined as:

$$\begin{aligned}\mathcal{L} = \lambda_1\mathcal{L}_{\text{class}} + \lambda_2\mathcal{L}_{\text{position}} + \lambda_3\mathcal{L}_{\text{rot}} + \lambda_4\mathcal{L}_{\text{size}} \\ + \lambda_5\mathcal{L}_{\text{depth}} + \lambda_6\mathcal{L}_{\text{normal}} + \lambda_7\mathcal{L}_{\text{seg}},\end{aligned} \tag{10}$$

where λ_1 to λ_7 are the weights to trade off different loss terms. $\mathcal{L}_{\text{position}}$, $\mathcal{L}_{\text{rot}}$ and $\mathcal{L}_{\text{size}}$ refer to loss terms for estimating 3D translation, rotation and metric size, respectively. Due to the space limitation, we refer readers to the *supplementary document* for more details on loss functions.

4 Experiments

4.1 Datasets and Evaluation metrics

Following [Manhardt *et al.*, 2020; Wei *et al.*, 2024; Zhang *et al.*, 2024], we trained and evaluated our proposed SeqPose on the **REAL** and **Context-Aware MixedEd ReAlity (CAMERA)** datasets [Wang *et al.*, 2019] to verify the effectiveness of our method on both video and non-video datasets. The **CAMERA** dataset is rendered from CAD models of 1,085 object instances against real backgrounds, it contains 300,000 synthetic images, of which 25,000 constitute the CAMERA25 dataset for testing. The **REAL** dataset captures 42 object instances in the real world and includes 18 videos under different real scenes. It comprises 4,300 images for training, 950 for validation, and 2,750 constitute the REAL275 for testing. The above datasets contain six categories including *bowl*, *bottle*, *can*, *camera*, *mug*, and *laptop*.

We adopt the commonly used metric, *i.e*. mean Average Precision (mAP) across all object categories to evaluate the performance. Specifically, we calculate the mAP by setting thresholds of 3D Intersection-Over-Union (3D IoU) at 25%, 50% and 75%, *i.e*. IoU25, IoU50 and IoU75, the 3D bounding box is derived from the estimated pose and size. We also compute the mAP by setting a rotation error and a translation error within specific degrees and centimeters, *i.e*. the mAP at 10°, 10cm, 10°10cm, 10°5cm and 5°5cm.

4.2 Implementation Details

We implemented SeqPose based on the DETR3D [Wang *et al.*, 2021b] framework. For each dataset, we trained our model with the same hyperparameter settings. Concretely, the model is trained for 24 epochs with batch size 16 and optimized using the AdamW [Loshchilov and Hutter, 2019] optimizer. The learning rate is initialized to 4e-4 and updated following a cosine annealing policy. We set all the loss weights in Eq. (10) as 1.0, except for λ_3=10.0 to penalize the rotation errors. The input size is 238×322. Moreover, we trained our model under single-frame (SF) and multi-frame (MF) settings on the **REAL** dataset, separately. For the **CAMERA** dataset, only the SF version of our model is trained, as the dataset exclusively consists of synthetic single-frame RGB images. Additionally, we set the frame length to three for memory queue updating under the MF setting. For the SF model, only 3D query anchors of the current frame are used to estimate the results. All experiments are implemented using PyTorch on a workstation equipped with four NVIDIA GeForce RTX 4090 GPUs. During testing, our model achieves an inference speed of 6.2 FPS. For more details on the implementation, please refer to our *supplementary document*.

4.3 Main Results

Quantitative results. We first compared our SeqPose with SOTA methods for RGB-based category-level object pose estimation on the CAMERA25 (Single-Frame) and REAL275 (Video) datasets, including Synthesis [Chen *et al.*, 2020b], CPS++ [Manhardt *et al.*, 2020], MSOS [Lee *et al.*, 2021], ODL-Net [Fan *et al.*, 2022], FAP-Net [Li *et al.*, 2024], LaPose [Zhang *et al.*, 2024] and DMSR [Wei *et al.*, 2024]. The quantitative results are listed in Table 1. It shows that our models can outperform SOTA methods in terms of 10°10cm and 5°5cm. In particular, for translation estimation in terms of mAP@10cm, our model achieves 26.0% and 35.9% improvements compared to the SOTA method DMSR [Wei *et al.*, 2024] on the REAL275 and CAMERA25, respectively. Moreover, our model also improves the performance on rotation estimation, achieving a significant performance gain of 5.4% and 0.9% in terms of mAP@10° on the REAL275 and CAMERA25, respectively. For 3D IoU metrics, our model also achieves competitive results. Concretely, the mAP@IoU50 of our SeqPose is 12.8% and 19.8% higher than DMSR on REAL275 and CAMERA25, respectively. Even for the strictest metric, *i.e*. IoU75, our models also realize 2.4% and 6.1% improvement compared with DMSR.

Figure 3 presents a detailed analysis in terms of Average Precision (AP) for each category against different types of error thresholds, including 3D IoU(%), rotation (°) and translation (cm). It shows that our model delivers considerable su-

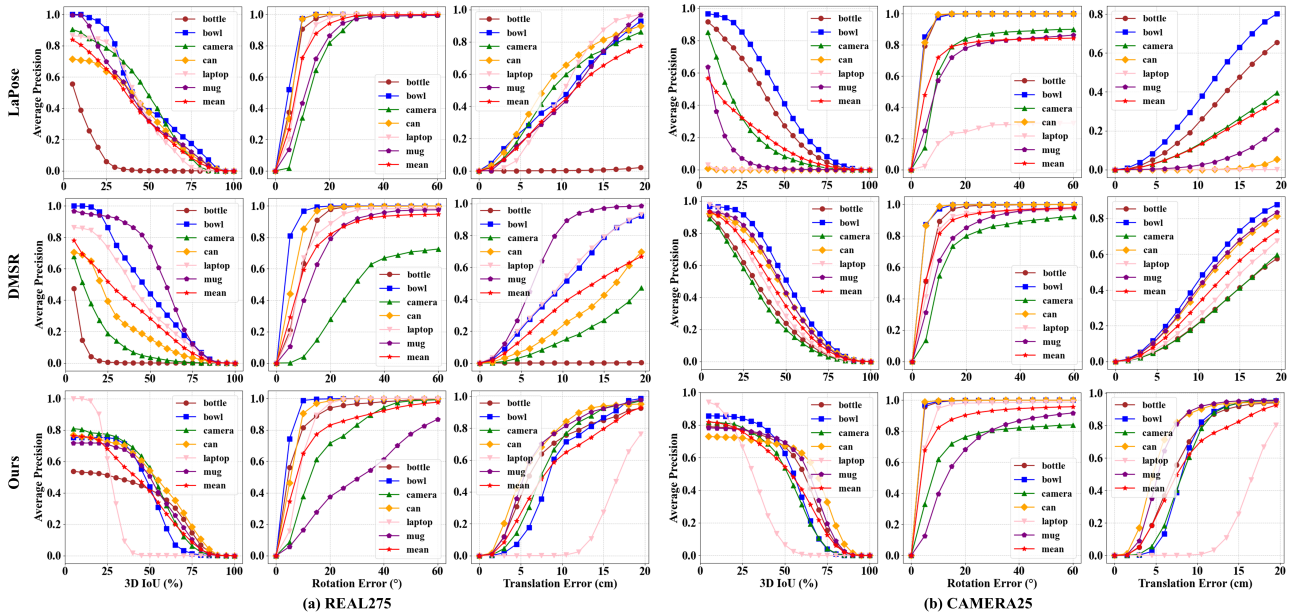
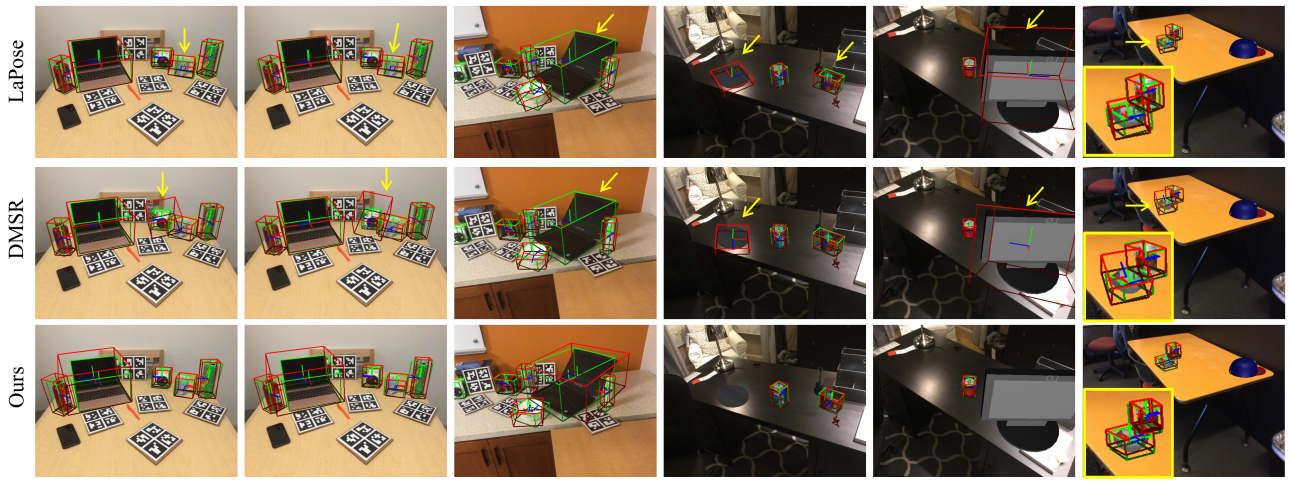


Figure 3: AP curves in terms of different types of errors on (a) REAL275 and (b) CAMERA25 datasets.



(a) REAL275 (b) CAMERA25

Figure 4: Qualitative results. The predicted and GT 3D bounding boxes are marked in red and green, respectively.

periority in comparison with the SOTA method DMSR w.r.t. different error thresholds across various object categories, especially for slender objects, *i.e.* "bottle", and complex-shaped objects, *i.e.* "camera" on both REAL and CAMERA datasets. We believe this is due to the powerful generalization ability of the DINOv2 and depth decoder. The robust semantic and geometry features can be generalized for objects with large intra-class texture and shape variations. On the other hand, our model indicates slightly poor AP values on the "laptop" class in terms of the 3D IoU and translation error. This may be related to the articulation of the "laptop" that exhibits more DoF in a pose with more flexible shapes. Since our model does not explicitly incorporate shape priors as constraints, the implicit 3D query anchors may be confused by the articulation states and variations of centroid/corner points. It is worth further investigation to apply our model to articulated objects leveraging kinematic constraints in future work.

Qualitative results. We further present Figure 4 to visualize the estimated object pose and size via projected 3D bounding boxes and object coordinate axes compared with the SOTA methods, *i.e.*, LaPose [Zhang *et al.*, 2024] and DMSR [Wei *et al.*, 2024]. Specifically, for each dataset, the first two examples are consecutive frames. The DMSR results for the "camera" class show a large shift in rotation and scale across different frames. In contrast, our model can produce more stable and consistent results by introducing spatial-temporal information. Moreover, the results of DMSR in the third, fourth and fifth columns indicate object instances

	IoU25	IoU50	IoU75	10cm	10°	10°10cm	10°5cm	5°5cm
EVAViT encoder	60.3	34.7	7.2	54.4	63.2	36.7	16.6	7.7
DINOv2 encoder	66.6	38.4	7.1	60.4	63.3	40	14.4	8.5
w/o Seg. Pred.	67.8	38.7	8.4	60.9	61.7	40	16.7	9
w/o Depth Pred.	67.8	39.5	**8.5**	61.1	64.3	41.9	16.9	**12.4**
w/o Norm. Pred.	67.1	39.1	7.8	62.3	63.2	**42**	16.4	11.8
w/o STSQ Memory	65.7	34.1	6.1	60.7	60.3	39.2	16.5	7.7
Ours (full)	**68.1**	**41.1**	**8.5**	**63.3**	**64.9**	41.8	**17.2**	10.4

Table 2: Ablation on the proposed modules.

are missing or wrongly located. It may related to the cumulative errors of 2D object detectors in multi-stage settings. In contrast, our model performs better with a higher recall for accurately locating objects. In particular, the poses and metric sizes of small objects shown in the first, second and last columns can be accurately estimated with tight 3D-oriented bounding boxes. Moreover, It also shows inaccurate pose and size estimation for the "laptop" and "mugs" classes. We refer the reader to the *supplementary document* for more visual examples, failure cases and discussion.

4.4 Ablation Study

Ablation on proposed modules. To further validate the effectiveness and contribution of the proposed modules, we performed several ablations to analyze our model on the REAL275 dataset in terms of the LVM backbone, auxiliary prediction heads and STSQ module, respectively. The quantitative results are listed in Table 2. Specifically, we trained baseline models by directly using *EVAViT encoder* [Li *et al.*, 2022], and *DINOv2 encoder* [Oquab *et al.*, 2023] without introducing the Depth decoder of DepthAnythingv2 [Yang *et al.*, 2024b]. For auxiliary prediction heads, models without segmentation head (*w/o Seg. Pred.*), depth prediction head (*w/o Depth Pred.*), and normal prediction head (*w/o Norm. Pred.*), are trained separately for verifying the effectiveness of different supervision for feature learning. The *w/o STSQ Memory* refers to the SeqPose model without introducing the spatial-temporal correlated sparse query memory queue.

Table 2 shows that the SeqPose with DINOv2 encoder achieved a better mAP than EVAViT. It can be treated as a strong baseline and demonstrates the effectiveness of the DINOv2 in extracting rich and robust semantic prior features. Moreover, for auxiliary prediction heads, the performance can be further improved by introducing extra supervision to learn semantic and geometry-aware features. In particular, the model fine-tuned without introducing the depth estimation head, *i.e*. *w/o Depth Pred.*, achieved better performance across all the metrics but a slightly lower mAP@10cm than other baseline models. For the STSQ module, it shows the model without introducing spatial-temporal correlated sparse query memory queue (*w/o STSQ Memory*) degrades the pose and size estimation performance in terms of all the metrics.

Our overall model achieved better performance than all the baseline models. It demonstrates the full model can benefit from: 1) The DINOv2 encoder produces robust intra-frame semantic features; 2) The auxiliary prediction heads can enhance the performance by encouraging the model to produce geometry-aware features compensating for intra-frame features; and 3) The spatial-temporal query memory queue further improve pose and size estimation performance by leveraging inter-frame feature interactions and correlations.

		IoU25	IoU50	IoU75	10cm	10°	10°10cm	10°5cm	5°5cm
Frames	0	65.7	34.1	6.1	60.7	60.3	39.2	16.5	7.7
	1	63	35.8	6.5	60.0	61.9	40.8	16.2	8.1
	2	64.5	36.0	6.7	60.4	61.8	40.1	14.9	9
	3	**68.1**	**41.1**	**8.5**	**63.3**	64.9	**41.8**	**17.2**	**10.4**
	4	66.6	38.4	7.1	60.4	63.3	40	14.4	8.5
	5	64.4	36.2	6.4	58.5	63.4	41.7	15.1	8.2
	6	65.8	36.7	6.7	59.8	**65.7**	40.5	15.5	9.3
	7	64	36.1	5.9	60.1	64.3	40.8	15.1	8.7
	8	63.3	36.0	7.5	60.0	63.4	**41.8**	17.1	9.8
Queries	300	58.1	31.9	6.2	60.0	62.7	39.1	15	8.2
	600	68.1	**41.1**	**8.5**	63.3	64.9	41.8	17.2	**10.4**
	900	66.8	38.9	8.2	**63.4**	62.1	**43.1**	**17.3**	9
	1200	**69.2**	39.8	**8.5**	60.0	**65.7**	40.2	16.6	9.9

Table 3: Ablation on frame length and the number of queries.

Ablation on the number of frames and queries. To further evaluate the effectiveness of the STSQ module, we performed another two sets of ablations on the number of frames in the query memory queue and the number of 3D anchor queries of the current frames on the REAL dataset. Quantitative results are listed in Table 3. Concretely, we ablate the performance of the STSQ memory queue by increasing the number of frames. Table 3 shows that the performance can be improved by increasing the number of frames to three, but degraded by setting more frames. It demonstrates the effectiveness of inter-frame query propagation by introducing temporal contextual relations. However, the performance may be degraded when the number of frames is larger than five. The reason may be related to the limitation of the STSQ module in capturing much longer temporal correlations for global range querying. Therefore, in our implementation, we set the number of historical frames as three in the memory queue.

Table 3 shows that the performance concerning pose and size estimation can be significantly improved when adding more implicit 3D anchors to 600, but slightly degraded by setting the number of 3D query anchors to 1200. Considering the trade-off between computational complexity and accuracy, we finally set the number of queries to 600.

5 Conclusion

We propose SeqPose, a novel end-to-end framework that unifies single-frame and video-based category-level object pose and size estimation. The DINOv2 encoder and depth decoder are incorporated to integrate intra-frame hierarchical semantic and geometric features. Furthermore, a spatial-temporal sparse query network is developed to propagate inter-frame correlations and aggregate intra-frame features by leveraging a set of learnable 3D query anchors. A pose prediction head is employed and optimized using the bipartite matching algorithm for set predictions. The whole framework is trainable in an end-to-end manner, enabling shape-prior-free pose and size estimation. Experimental results demonstrate our model can outperform SOTA methods in both single-frame and video-based categorical pose estimation on the CAMERA and REAL datasets. In future work, it would be valuable for further investigation to generalize our model for challenging objects and scenes, *e.g*., articulated and unseen objects.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China (NSFC) under grants: 62302104 and 62476063, the National Key Research and Development Program of China under Grant 2025YFE0101100, the Natural Science Foundation of Guangdong Province under grants: 2023A1515012884 and 2025A1515011293, and the Science and Technology Program of Guangzhou under grant: SL2023A04J01625.

References

[Carion *et al.*, 2020] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, volume 12346, pages 213–229. Springer, 2020.

[Caron *et al.*, 2021] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *ICCV*, pages 9630–9640. IEEE, 2021.

[Chen and Dou, 2021] Kai Chen and Qi Dou. SGPA: structure-guided prior adaptation for category-level 6d object pose estimation. In *ICCV*, pages 2753–2762. IEEE, 2021.

[Chen *et al.*, 2020a] Dengsheng Chen, Jun Li, Zheng Wang, and Kai Xu. Learning canonical shape space for category-level 6d object pose and size estimation. In *CVPR*, pages 11970–11979. IEEE, 2020.

[Chen *et al.*, 2020b] Xu Chen, Zijian Dong, Jie Song, Andreas Geiger, and Otmar Hilliges. Category level object pose estimation via neural analysis-by-synthesis. In *ECCV*, volume 12371, pages 139–156. Springer, 2020.

[Chen *et al.*, 2024] Yamei Chen, Yan Di, Guangyao Zhai, Fabian Manhardt, Chenyangguang Zhang, Ruida Zhang, Federico Tombari, Nassir Navab, and Benjamin Busam. Secondpose: Se(3)-consistent dual-stream feature fusion for category-level pose estimation. In *CVPR*, pages 9959–9969. IEEE, 2024.

[Fan *et al.*, 2022] Zhaoxin Fan, Zhenbo Song, Jian Xu, Zhicheng Wang, Kejian Wu, Hongyan Liu, and Jun He. Object level depth reconstruction for category level 6d object pose estimation from monocular RGB image. In *ECCV*, volume 13662, pages 220–236. Springer, 2022.

[He *et al.*, 2017] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. Mask R-CNN. In *ICCV*, pages 2980–2988. IEEE, 2017.

[Hoque *et al.*, 2023] Sabera Hoque, Shuxiang Xu, Ananda Maiti, Yuchen Wei, and Md. Yasir Arafat. Deep learning for 6d pose estimation of objects - A case study for autonomous driving. *Expert Syst. Appl.*, 223:119838, 2023.

[Lee *et al.*, 2021] Taeyeop Lee, Byeong-Uk Lee, Myungchul Kim, and In So Kweon. Category-level metric scale object shape and pose estimation. *IEEE Robotics Autom. Lett.*, 6(4):8575–8582, 2021.

[Lepetit *et al.*, 2009] Vincent Lepetit, Francesc Moreno-Noguer, and Pascal Fua. Ep*n*p: An accurate $O(n)$ solution to the p*n*p problem. *Int. J. Comput. Vis.*, 81(2):155–166, 2009.

[Li *et al.*, 2022] Yanghao Li, Hanzi Mao, Ross B. Girshick, and Kaiming He. Exploring plain vision transformer backbones for object detection. In *ECCV*, volume 13669, pages 280–296. Springer, 2022.

[Li *et al.*, 2024] Jia Li, Li Jin, Xibin Song, Yeheng Chen, Nan Li, and Xueying Qin. Implicit coarse-to-fine 3d perception for category-level object pose estimation from monocular RGB image. In *ICRA*, pages 2043–2050. IEEE, 2024.

[Lin *et al.*, 2024] Xiao Lin, Minghao Zhu, Ronghao Dang, Guangliang Zhou, Shaolong Shu, Feng Lin, Chengju Liu, and Qijun Chen. Clipose: Category-level object pose estimation with pre-trained vision-language knowledge. *IEEE Trans. Circuits Syst. Video Technol.*, 34(10):9125–9138, 2024.

[Liu *et al.*, 2023a] Chongpei Liu, Wei Sun, Jian Liu, Xing Zhang, Shimeng Fan, and Qiang Fu. Fine segmentation and difference-aware shape adjustment for category-level 6dof object pose estimation. *Appl. Intell.*, 53(20):23711–23728, 2023.

[Liu *et al.*, 2023b] Jian Liu, Wei Sun, Chongpei Liu, Xing Zhang, and Qiang Fu. Robotic continuous grasping system by shape transformer-guided multiobject category-level 6-d pose estimation. *IEEE Trans. Ind. Informatics*, 19(11):11171–11181, 2023.

[Liu *et al.*, 2023c] Jianhui Liu, Yukang Chen, Xiaoqing Ye, and Xiaojuan Qi. Ist-net: Prior-free category-level pose estimation with implicit space transformation. In *ICCV*, pages 13932–13942. IEEE, 2023.

[Liu *et al.*, 2023d] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, and Lei Zhang. Grounding DINO: marrying DINO with grounded pre-training for open-set object detection. *CoRR*, abs/2303.05499, 2023.

[Liu *et al.*, 2024] Jian Liu, Wei Sun, Hui Yang, Zhiwen Zeng, Chongpei Liu, Jin Zheng, Xingyu Liu, Hossein Rahmani, Nicu Sebe, and Ajmal Mian. Deep learning-based object pose estimation: A comprehensive survey. *CoRR*, abs/2405.07801, 2024.

[Loshchilov and Hutter, 2019] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*. OpenReview.net, 2019.

[Manhardt *et al.*, 2020] Fabian Manhardt, Gu Wang, Benjamin Busam, Manuel Nickel, Sven Meier, Luca Minciullo, Xiangyang Ji, and Nassi Navab. Cps++: Improving class-level 6d pose and shape estimation from monocular images with self-supervised learning. *CoRR*, abs/2003.05848, 2020.

[Oquab *et al.*, 2023] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa,

Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael G. Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jégou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision. *CoRR*, abs/2304.07193, 2023.

[Su *et al.*, 2019] Yongzhi Su, Jason R. Rambach, Nareg Minaskan, Paul Lesur, Alain Pagani, and Didier Stricker. Deep multi-state object pose estimation for augmented reality assembly. In *ISMAR*, pages 222–227. IEEE, 2019.

[Sun *et al.*, 2022] Jingtao Sun, Yaonan Wang, Mingtao Feng, Danwei Wang, Jiawen Zhao, Cyrill Stachniss, and Xieyuanli Chen. Ick-track: A category-level 6-dof pose tracker using inter-frame consistent keypoints for aerial manipulation. In *IROS*, pages 1556–1563. IEEE, 2022.

[Sun *et al.*, 2023] Han Sun, Peiyuan Ni, Zhiqi Li, Yizhao Wang, Xiaoxiao Zhu, and Qixin Cao. Panelpose: A 6d pose estimation of highly-variable panel object for robotic robust cockpit panel inspection. In *IROS*, pages 3214–3221, 2023.

[Sun *et al.*, 2024] Jingtao Sun, Yaonan Wang, Mingtao Feng, Yulan Guo, Ajmal Mian, and Mike Zheng Shou. L4d-track: Language-to-4d modeling towards 6-dof tracking and shape reconstruction in 3d point cloud stream. In *CVPR*, pages 21146–21156. IEEE, 2024.

[Tian *et al.*, 2020] Meng Tian, Marcelo H. Ang, and Gim Hee Lee. Shape prior deformation for categorical 6d object pose and size estimation. In *ECCV*, volume 12366, pages 530–546. Springer, 2020.

[Umeyama, 1991] Shinji Umeyama. Least-squares estimation of transformation parameters between two point patterns. *IEEE Trans. Pattern Anal. Mach. Intell.*, 13(4):376–380, 1991.

[Wang *et al.*, 2019] He Wang, Srinath Sridhar, Jingwei Huang, Julien Valentin, Shuran Song, and Leonidas J. Guibas. Normalized object coordinate space for category-level 6d object pose and size estimation. In *CVPR*, pages 2642–2651. IEEE, 2019.

[Wang *et al.*, 2021a] Jiaze Wang, Kai Chen, and Qi Dou. Category-level 6d object pose estimation via cascaded relation and recurrent reconstruction networks. In *IROS*, pages 4807–4814. IEEE, 2021.

[Wang *et al.*, 2021b] Yue Wang, Vitor Guizilini, Tianyuan Zhang, Yilun Wang, Hang Zhao, and Justin Solomon. DETR3D: 3d object detection from multi-view images via 3d-to-2d queries. In *CoRL*, volume 164 of *Proceedings of Machine Learning Research*, pages 180–191. PMLR, 2021.

[Wang *et al.*, 2023] Shihao Wang, Yingfei Liu, Tiancai Wang, Ying Li, and Xiangyu Zhang. Exploring object-centric temporal modeling for efficient multi-view 3d object detection. In *ICCV*, pages 3598–3608. IEEE, 2023.

[Wang *et al.*, 2024] Pengyuan Wang, Takuya Ikeda, Robert Lee, and Koichi Nishiwaki. Gs-pose: Category-level object pose estimation via geometric and semantic correspondence. In Ales Leonardis, Elisa Ricci, Stefan Roth, Olga Russakovsky, Torsten Sattler, and Gül Varol, editors, *ECCV*, volume 15085, pages 108–126. Springer, 2024.

[Wei *et al.*, 2024] Jiaxin Wei, Xibin Song, Weizhe Liu, Laurent Kneip, Hongdong Li, and Pan Ji. Rgb-based category-level object pose estimation via decoupled metric scale recovery. pages 2036–2042, 2024.

[Wen *et al.*, 2023] Bowen Wen, Wei Yang, Jan Kautz, and Stan Birchfield. Foundationpose: Unified 6d pose estimation and tracking of novel objects. *CoRR*, abs/2312.08344, 2023.

[Yang *et al.*, 2024a] Lihe Yang, Bingyi Kang, Zilong Huang, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything: Unleashing the power of large-scale unlabeled data. *CoRR*, abs/2401.10891, 2024.

[Yang *et al.*, 2024b] Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything V2. *CoRR*, abs/2406.09414, 2024.

[Yu *et al.*, 2023] Sheng Yu, Di-Hua Zhai, Yuyin Guan, and Yuanqing Xia. Category-level 6-d object pose estimation with shape deformation for robotic grasp detection. *IEEE Transactions on Neural Networks and Learning Systems*, 2023.

[Zhang *et al.*, 2024] Ruida Zhang, Ziqin Huang, Gu Wang, Chenyangguang Zhang, Yan Di, Xingxing Zuo, Jiwen Tang, and Xiangyang Ji. Lapose: Laplacian mixture shape modeling for rgb-based category-level object pose estimation. In *ECCV*, volume 15083 of *Lecture Notes in Computer Science*, pages 467–484. Springer, 2024.

[Zheng *et al.*, 2024] Linfang Zheng, Tze Ho Elden Tse, Chen Wang, Yinghan Sun, Hua Chen, Ales Leonardis, Wei Zhang, and Hyung Jin Chang. Georef: Geometric alignment across shape variation for category-level object pose refinement. In *CVPR*, pages 10693–10703. IEEE, 2024.

[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*, pages 5745–5753. IEEE, 2019.

[Zhu *et al.*, 2021] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable DETR: deformable transformers for end-to-end object detection. In *ICLR*. OpenReview.net, 2021.