IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

DynOPETs: A Versatile Benchmark for Dynamic Object Pose Estimation and Tracking in Moving Camera Scenarios

Xiangting Meng*[1], Jiaqi Yang*[1], Mingshu Chen[2], Chenxin Yan[1], Yujiao Shi[1], Wenchao Ding[2†], Laurent Kneip[1,3†]

***Abstract*—In the realm of object pose estimation, scenarios involving both dynamic objects and moving cameras are prevalent. However, the scarcity of corresponding real-world datasets significantly hinders the development and evaluation of robust pose estimation models. This is largely attributed to the inherent challenges in accurately annotating object poses in dynamic scenes captured by moving cameras. To bridge this gap, this paper presents a novel dataset *DynOPETs* and a dedicated data acquisition and annotation pipeline tailored for object pose estimation and tracking in such unconstrained environments. Our efficient annotation method innovatively integrates pose estimation and pose tracking techniques to generate pseudo-labels, which are subsequently refined through pose graph optimization. The resulting dataset offers accurate pose annotations for dynamic objects observed from moving cameras. To validate the effectiveness and value of our dataset, we perform comprehensive evaluations using 19 state-of-the-art methods, demonstrating its potential to accelerate research in this challenging domain. The dataset will be made publicly available to facilitate further exploration and advancement in the field.**

Fig. 1. The full set of dynamic objects in our dataset.

I. INTRODUCTION

OBJECT pose estimation, which involves determining an object's 6-DoF pose (position and orientation) in 3D space, is a fundamental research challenge in computer vision and robotics. The task is crucial for various applications in embodied intelligence, augmented reality (AR), and mixed reality (MR) systems [1], such as robotic manipulation [2], dexterous hand control, and human-object interaction [3]. Despite significant advancements in recent years driven by the emergence of benchmark datasets, existing real-world datasets exhibit notable limitations. Most of the existing datasets predominantly assume static conditions - either fixed camera positions or stationary objects. This assumption contrasts sharply with real-world requirements where simultaneous camera egomotion and object dynamics must be addressed.

The limited availability of dynamic pose estimation datasets stems primarily from annotation challenges inherent to moving-camera-dynamic-object configurations. While static-object scenarios allow for indirect pose annotation via the state-of-the-art camera pose estimation technique, such as SLAM algorithm [4] or the fiducial marker-based localization method [5], dynamic settings require direct object motion tracking. In such dynamic settings, existing solutions for pose annotation rely on either labour-intensive manual annotation [6] or expensive motion capture (MoCap) systems in controlled environments [7]. Another approach [8] involves using state-of-the-art object pose estimation methods [1] to generate high-quality pseudo-labels for annotation. However, many common objects in daily life, such as those with symmetric geometries, transparent or reflective materials, or sparse/repetitive patterns, pose significant challenges for those algorithms. As a result, a significant fraction of the object pose annotations still require manual refinement due to the low quality of the pseudo-labels.

To address the issue of scarce real-world datasets in this challenging scenario, we propose a novel dataset that includes diverse object categories and dynamic scenes. The dataset primarily consists of RGB-D video sequences, object CAD models, and high-quality object and camera pose annotations, supporting various tasks in the field of object pose estimation. To tackle the challenges of pose annotation, we introduce a novel automated pose annotation pipeline leveraging the fusion of absolute and relative object pose estimates, significantly improving data labeling efficiency and reducing human effort. Finally, we systematically investigate existing state-of-the-art methods and comprehensively benchmark their performance in

† *Corresponding Authors: Wenchao Ding and Laurent Kneip.*

[1] Xiangting Meng, Jiaqi Yang, Chenxin Yan, Yujiao Shi, and Laurent Kneip are with the Mobile Perception Lab, School of Information Science and Technology, ShanghaiTech University. (e-mail: mengxt@shanghaitech.edu.cn; yangjq12022@shanghaitech.edu.cn; yanchx1@shanghaitech.edu.cn; shiyj2@shanghaitech.edu.cn; lkneip@shanghaitech.edu.cn)

[2] Mingshu Chen and Wenchao Ding are with the Multi-Agent Robotic Systems Lab, Academy for Engineering and Technology, Fudan University. (e-mail: chenms24@m.fudan.edu.cn; dingwenchao@fudan.edu.cn)

[3] Laurent Kneip is also with the Shanghai Engineering Research Center of Intelligent Vision and Imaging.

* Authors contributed equally to this work.

The code implementation, along with the dataset and video, can be found at the project page: https://stay332.github.io/DynOPETs.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

dynamic scenarios, demonstrating the value of our work. We believe that our proposed dataset and efficient pose annotation pipeline can fill the gap in dynamic object pose estimation datasets, thereby promoting the development of applications in the fields of embodied intelligence and AR/MR. Our main contributions are summarized as follows:

- We propose an efficient pose annotation approach that integrates an absolute pose estimator with a global KF smoother, a relative pose estimator based on point tracking, and an object pose graph optimization module. This pipeline enables accurate, marker-free dynamic object pose labeling, without extensive manual effort.
- We present *DynOPETs*, a comprehensive object pose estimation dataset comprising RGB-D sequences of 175 distinct object instances undergoing simultaneous camera and object motion, with synchronized 6-DoF pose annotations for both camera and object trajectories.
- We conduct a systematic benchmark of various taxonomy-specific pose estimation and object pose tracking methods on our dataset.

II. RELATED WORK

According to [19], we categorize object pose estimation methods into 4 classes: instance-level object pose estimation (IOPE), unseen object pose estimation (UOPE), category-level object pose estimation (COPE), and object pose tracking (OPT). This section discusses representative and state-of-the-art methods within each class, followed by an overview of recently published datasets relevant to these areas.

A. Object Pose Estimation and Tracking Methods

1) Instance-level Object Pose Estimation: These methods predict the pose of only those objects present in the training set, often relying on priors such as CAD models or posed reference views during inference [20]. Object pose estimation can be achieved through template matching [21] or feature correspondence matching [22] given these priors. Some approaches directly predict object pose using neural networks [20], while others employ voting schemes [23]. Instance-level methods are the most mature but also the most restrictive due to their inability to generalize to novel objects. Consequently, research has shifted towards the more generalizable UOPE and COPE methods, which are the primary focus of this paper and our proposed dataset.

2) Unseen Object Pose Estimation: Similar to IOPE methods, UOPE methods rely on CAD models or posed reference views as priors. However, UOPE methods can generalize to novel objects, enabling pose prediction for unseen objects during inference. Recent works in this area are broadly categorized into two main strategies: feature matching and template matching-based methods. Feature matching methods [24], [25], [26], [27] extract robust image features and establish correspondences with generic, object-agnostic CAD model features. These approaches enable zero-shot estimation but may rely heavily on feature discriminability. In contrast, template matching methods [1], [28], [29], [30] leverage extensive sets of object templates to achieve strong generalization across novel objects.

3) Category-level Object Pose Estimation: COPE methods typically handle novel objects within the same category by learning category-specific shape representations, eliminating the reliance on CAD models and reference views. NOCS [9] pioneered this approach by introducing the normalized object coordinate space (NOCS), a canonical representation where all objects within the same category are consistently aligned in terms of size and orientation. Aligning predicted canonical representations with depth points allows for the estimation of object pose.

Recent advancements have explored different network designs, such as using vision transformers [31] or diffusion models [32] to predict better canonical representations. Several works [32], [33], [34], [35], [36] employ deformed categorical shape templates to improve canonical representations, while [11], [37], [38], [39] formulate COPE as a generative task. Furthermore, some methods directly predict object pose, employing either a combination of geometric and semantic priors [40], [41], [42] or enhanced geometric feature extractors [43].

4) Object Pose Tracking: Unlike single-frame object pose estimation methods, object pose tracking leverages temporal information across video sequences to estimate object poses [19]. Typically, tracking methods also require posed reference views or CAD models. While classical methods [44] often track object poses using geometric or feature-based approaches, recent methods [45] jointly track and refine object shape through neural implicit representations, providing robust performance. Moreover, [1] leverages large-scale pre-trained models and efficiently tracks unseen objects using large-scale synthetic training data for enhanced generalization.

B. Object Pose Estimation Datasets

Object pose estimation datasets have evolved significantly over the years. We show a dataset comparison in table I. The following is a brief overview.

Linemod [14] introduced a standard instance-level RGB-D benchmark featuring texture-less objects with limited occlusions in relatively simple scenarios. YCB-Video [12] subsequently expanded realism with household objects. T-Less [13] introduced texture-less but symmetric objects in challenging industrial scenarios, highlighting pose ambiguity issues. REAL275 [9] established the first real-world benchmark for category-level pose estimation by introducing canonical object representations across 6 categories. StereOBJ-1M [15] further tackled challenging materials with a much larger scale, such as transparent and reflective objects. To further diversify modalities, PhoCaL [17] provided multimodal images focusing on challenging transparent and reflective household items with 8 categories. Wild6D [16] advanced realism further by capturing large-scale videos of objects in the wild. More recently, HANDAL [8] focused on graspable objects, collecting 17 categories of manipulable items in cluttered scenes via multi-view imagery and annotating poses with [45]. HOT3D [3] was the first to address egocentric scenarios featuring dense hand–object interactions. However, it incorporates a limited variety of objects and constrained scene settings. HouseCat6D [18] delivered highly accurate annotations for 10 categories of household

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

TABLE I

COMPARISON BETWEEN EXISTING 17 DATASETS WITH OUR PROPOSED DATASET. THE TABLE SUMMARIZES KEY ATTRIBUTES USED FOR COMPARING DATASETS: OBJECT CATEGORIES, WHETHER THE DATA IS SYNTHETIC OR REAL-WORLD, DATA MODALITIES (RGB, DEPTH), OBJECT DETAILS (NUMBER OF INSTANCES, PRESENCE OF DYNAMIC OBJECTS, PRESENCE OF MOVING CAMERAS, CAD MODEL AVAILABILITY), WHETHER THE VIDEOS ARE MARKER-FREE, AND DATASET SIZE (NUMBER OF VIDEOS, IMAGES, AND ANNOTATIONS).

Dataset	Cat.	Real	Modality		Object Details				Marker-Free	Vid.	Img.	Anno.
			RGB	Depth	Num.	CAD	Dyna-Obj	Mov-Cam				
CAMERA25 [9]	6	✗	✓	✓	1085	✓	✗	✓	✓	-	300K	4M
Omni6D [10]	166	✗	✓	✓	4688	✓	✗	✓	✓	-	0.8M	5.6M
SOPE [11]	149	✗	✓	✓	4162	✓	✗	✓	✓	-	475K	5M
YCB-Video [12]	-	✓	✓	✓	21	✓	✗	✓	✓	92	133K	0.6M
T-LESS [13]	-	✓	✓	✓	30	✓	✗	✓	✗	20	48K	48K
Linemod [14]	-	✓	✓	✓	15	✓	✗	✗	✗	-	18K	15k
StereoOBJ-1M [15]	-	✓	✗	✓	18	✓	✗	✓	✗	182	393K	1.5M
YCBInEOAT [6]	-	✓	✓	✓	5	✓	✓	✗	✓	9	7449	3.7K
HOT3D [3]	-	✓	✓	✗	33	✓	✓	✓	✗	425	3.7M	6M
UOPE-56 (Ours)	-	✓	✓	✓	56	✓	✓	✓	✓	56	64K	180K
REAL275 [9]	6	✓	✓	✓	42	✓	✗	✓	✗	18	8K	35K
Wild6D [16]	5	✓	✓	✓	162	✗	✗	✓	✓	486	10K	10K
PhoCaL [17]	8	✓	✓	✓	60	✓	✗	✓	✗	24	3.9K	91K
HANDAL [8]	17	✓	✓	✗	212	✓	✓	✗	✓	2.2K	308K	1.2M
HouseCat6D [18]	10	✓	✓	✓	194	✓	✗	✓	✓	41	23.5K	160K
ROPE [11]	149	✓	✓	✓	581	✓	✗	✓	✓	363	332K	1.5M
COPE-119 (Ours)	6	✓	✓	✓	119	✓	✓	✓	✓	119	140K	350K

objects using polarization imagery to enhance robustness to reflections. OmniNOCS [31] is a unified NOCS dataset that integrated data from multiple domains and featured over 90 object classes, making it the largest NOCS dataset to date. Omni6D [10] and ROPE [11] also provided a large-scale dataset for object pose estimation.

III. DATA ACQUISITION

A. Overview

The proposed *DynOPETs* consists of 175 RGB-D video sequences of distinct everyday objects captured on typical desktop backgrounds with sufficient visual features. Each sequence comprises roughly 500 frames recorded at 15 Hz, totaling approximately 210K RGB-D frames. Additionally, we provide over 530K annotations, including instance segmentations, camera intrinsic and extrinsic parameters, object poses, and camera poses. The dataset is divided into two subsets. The first, COPE-119, comprises 119 sequences of objects belonging to 6 categories commonly found in the COPE benchmark: bottles, bowls, cameras, cans, laptops, and mugs. The second, UOPE-56, contains 56 sequences of other common household objects (illustrated in fig. 1) and is designed for evaluating the performance of UOPE algorithms.

B. Hardware Setup and Objects

RGB-D data was captured using a consumer-grade iPad Pro with the hardware-synchronized Structure Sensor Pro [46] depth camera. CAD models for all 175 objects were scanned using the sensors' integrated RGB-D 3D modeling software. Following mesh reconstruction, objects were manually aligned so that their centers coincided with the origin of the 3D coordinate system, and their front, top, and side views aligned with the x, y, and z axes, respectively. Consistent with the definition in [9], the coordinate system of an object's CAD model is defined as Object Local Coordinates (OLC) system, also known as the canonical space in the COPE domain, as illustrated in Figure 3. Furthermore, we used SAM2 [47] to automatically segment objects in these frames.

C. Sensor Calibration and Synchronization

Intrinsic and extrinsic camera parameters were calibrated using the depth camera SDK, which provides simultaneous calibration of both RGB and depth cameras. We developed a custom camera driver based on the SDK that warps the depth map to the RGB camera's coordinate frame, resulting in co-registered RGB-D video sequences. A MoCap system provided camera poses. The relative transformation between the MoCap marker frame and the camera's optical center was calibrated using a hand-eye calibration procedure, yielding camera poses within the MoCap coordinate system. A temporal offset existed between the MoCap system's timestamps and the RGB-D frame timestamps. To resolve this, we used visual odometry (VO) [4] to estimate the camera pose for each

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

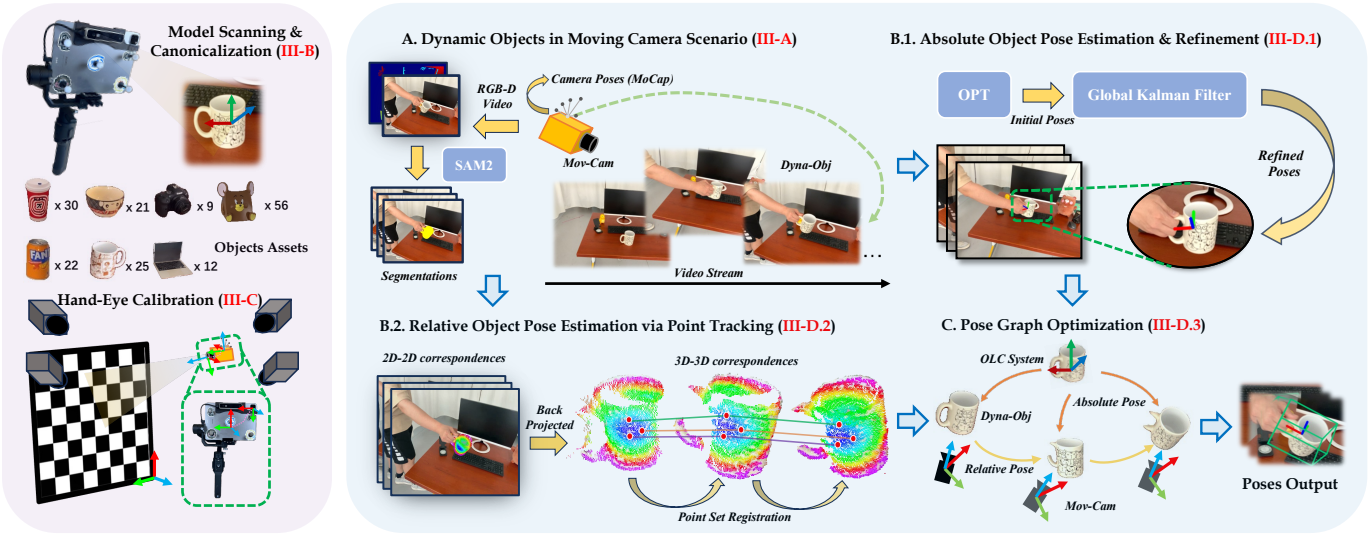

Fig. 2. Overview. The left block is CAD model scanning and preprocessing, as well as hand-eye calibration. The right block is the data acquisition and efficient object pose annotation pipeline. The dynamic object is captured by a moving RGB-D camera, and the ground truth object pose annotation is calculated by fusing refined absolute object pose estimates and relative object pose estimates. The camera pose is provided by a MoCap.

frame and registered these VO-derived poses with the MoCap poses. This registration process established a mapping between each camera frame and the corresponding MoCap timestamp.

D. Efficient Object Pose Annotation

Illustrated in fig. 2, to provide context for our approach, we begin with key definitions. Firstly, the absolute object pose, which we aim to annotate, is defined as the Sim(3) transformation (scale, rotation, and translation) required to align an object from its OLC system to the camera frame. Since our CAD models and corresponding point clouds are scaled to real-world dimensions, this Sim(3) simplifies to an SE(3) transformation (rotation and translation). Secondly, the relative object pose describes the SE(3) transformation between the object's poses at two different time steps in the camera coordinate system. These two definitions are directly relevant to the following two approaches for obtaining object pose annotations.

1) Absolute Object Pose Estimation and Refinement: The first approach involves directly generating high-quality pseudo-pose annotations using existing state-of-the-art algorithms. The model-based method [1] stands out in this category, leveraging CAD models to output object motion. Although generally accurate, [1] can occasionally generate discontinuous pose estimates. To address this issue, we adopt a Global Kalman Filter [48] (Global KF) framework with the Extended Kalman Filter (EKF) [48] and the Rauch-Tung-Striebel (RTS) smoother [49] to refine the object pose predictions, mitigating pose jitters. The RTS smoother employs a constant velocity model for both translation $p \in \mathbf{R}^3$ and rotational quaternion $q \in \mathbb{S}^3$, where state variables are defined as $\mathbf{x} = [p, q, v, \omega]$, where $v \in \mathbb{R}^3, \omega \in \mathbb{R}^3$ refers to the velocity and the angular velocity. We first perform a forward pass eq. (2) using the standard EKF prediction and update equations to obtain filtered state estimates $\mathbf{x}_t$ from step $t-1$ to step t, then, a backward pass eq. (3) using the RTS smoother refines these estimates from step $t+1$ to t:

$$\hat{\mathbf{x}}_t = \mathbf{F}\mathbf{x}_{t-1}, \tag{1}$$

$$\mathbf{x}_t = \hat{\mathbf{x}}_t + \mathbf{K}_t\left(\mathbf{z}_t - \mathbf{H}\hat{\mathbf{x}}_t\right), \tag{2}$$

$$\mathbf{x}_t = \mathbf{x}_t + \mathbf{G}_t(\mathbf{x}_{t+1} - \hat{\mathbf{x}}_{t+1}), \tag{3}$$

$$\mathbf{G}_t = \mathbf{P}_t\mathbf{F}^T\mathbf{P}_{t+1}^{-1}, \tag{4}$$

where $\mathbf{z}$ refers to the observed pose, and $\mathbf{F}$ and $\mathbf{H}$ represent the state transition and observation matrices, respectively. K is the extended Kalman gain used in the update step [48], $\mathbf{P}$ denotes the covariance matrix and $\mathbf{G}$ represents the smoothing gain. After our refinement, the pose annotations of objects with rich textures and distinctive shapes are now accurate and smooth, while for those objects exhibiting significant symmetry or lacking distinctive texture, such as objects within the bowl category, the pose annotation is still not high-quality enough.

2) Relative Object Pose Estimation via Point Tracking: To address the challenges posed by these objects, we explore a second approach that involves first estimating the absolute object pose in the initial frame and then iteratively computing the frame-to-frame relative object pose to derive the pose annotation for the entire sequence.

[44] serves as a representative example of this approach, employing keypoint detection and matching to establish 2D correspondences. These 2D correspondences are then back-projected into 3D space using sensor depths, forming 3D-3D correspondences that enable the computation of relative object poses via point-registration techniques. Our method adopts a similar strategy, but replaces the keypoint detection and matching component with a state-of-the-art Track Any Point (TAP) model [50]. TAP models accept a video sequence as input and output 2D correspondence points for a group of specified query points in the reference frame across all frames. Compared to traditional feature extraction and matching methods, the TAP model [50] is better equipped to leverage the

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

temporal information inherent in video sequences, enabling it to effectively handle objects with limited texture. While this method produces smoother object pose estimates and is more robust to symmetry ambiguities, it is susceptible to cumulative errors, leading to drift in the estimated trajectory over time.

3) Object Pose Graph Optimization: Recognizing the complementary nature of the two annotation methods discussed above, we construct a pose graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ to integrate them into a unified framework. The graph consists of nodes $\mathcal{V}$ and edges $\mathcal{E}$(illustrated in fig. 3). Each node $i \in \mathcal{V}$ represents the pose of an object at a specific time step, denoted as $\mathbf{T}_i \in SE(3)$, where $\mathbf{T}_i$ represents the transformation from the object's local coordinate system to the camera coordinate system at time i. In essence, objects in OLC and their corresponding camera frames at each time step become the nodes in this graph.

The edges $\mathcal{E}$ represent the constraints between these object poses. We utilize two types of edges: absolute pose edges and relative pose edges. Absolute pose edges, denoted as $(\text{OLC}, i) \in \mathcal{E}$, are derived from [1], refined by the Global KF framework in section III-D1. Relative pose edges, denoted as $(i, j) \in \mathcal{E}$, connect two nodes representing the same object at different time steps. These are derived from the point tracking method (TAP). We use $\mathbf{z}_{\text{OLC},i}, \mathbf{z}_{i,j} \in SE(3)$ to represent the associated measurements of these edges, and $\mathbf{\Omega}_{i,j}$ and $\mathbf{\Omega}_{\text{OLC},i}$ for the corresponding information matrices. Note that in our implementation, $\mathbf{\Omega}_{\text{OLC},i}$ is constant and $\mathbf{\Omega}_{i,j}$ are calculated from the Jacobian of the point set registration. Also, absolute pose edges with poor measurements can be removed manually by visual checks. See PDF supplementary material for details.

By encoding both absolute and relative poses as edges connecting these nodes, we form a graph with closed loops. Then we need to find the set of object poses $\mathbf{T} = \{\mathbf{T}_i\}_{i \in \mathcal{V}}$ that minimizes the following cost function:

$$\mathcal{F}(\mathbf{T}) = \sum_{(\text{OLC},i) \in \mathcal{E}} \mathbf{r}^T_{\text{OLC},i}(\mathbf{T}) \mathbf{\Omega}_{\text{OLC},i} \mathbf{r}_{\text{OLC},i}(\mathbf{T}) \tag{5}$$

$$+ \sum_{(i,j) \in \mathcal{E}} \mathbf{r}^T_{i,j}(\mathbf{T}) \mathbf{\Omega}_{i,j} \mathbf{r}_{i,j}(\mathbf{T}), \tag{6}$$

where $\mathbf{r}_{\text{OLC},i}(\mathbf{T})$ and $\mathbf{r}_{i,j}(\mathbf{T})$ are the residual error terms for absolute and relative pose edges, respectively, defined as:

$$\mathbf{r}_{\text{OLC},i}(\mathbf{T}) = \log(\mathbf{z}^{-1}_{\text{OLC},i} \mathbf{T}_i), \tag{7}$$

$$\mathbf{r}_{i,j}(\mathbf{T}) = \log(\mathbf{z}^{-1}_{i,j} \mathbf{T}^{-1}_i \mathbf{T}_j), \tag{8}$$

Here, $\log(\cdot)$ denotes the matrix logarithm, mapping an $SE(3)$ transformation to its Lie algebra representation.

The optimization problem, therefore, is to find:

$$\mathbf{T}^* = \arg\min_{\mathbf{T}} \mathcal{F}(\mathbf{T}). \tag{9}$$

This non-linear least squares problem can be solved using Gauss-Newton or Levenberg-Marquardt algorithms. By minimizing the cost function, we obtain refined object poses that are globally consistent, leveraging the strengths of both absolute and relative pose estimation methods. The manual effort required by our method is primarily limited to identifying and removing inaccurate absolute pose estimates, which minimizes human intervention and promotes annotation efficiency. It is worth noting that, overall, the Global KF framework yields the largest accuracy gains for the annotated object poses, while the relative pose estimation and pose-graph optimization stages mainly mitigate the impact of frames whose absolute poses are inaccurate.

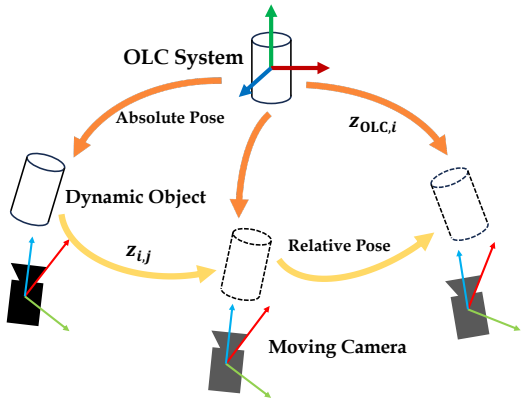

Fig. 3. Pose graph Optimization. The orange arrows represent absolute pose edges (OLC, i) and the yellow arrows represent relative pose edges (i, j). In the OLC space, the object's center aligns with the origin and has a semantically meaningful orientation. For example, the opening of a cup or bottle aligns with the y-axis, while the front and sides of the object are aligned with the x-axis and z-axis, respectively.

IV. EVALUATION

A. Comparisons with Pose from Motion Capture System

Before conducting benchmarks, we need to verify the accuracy of the pose labels obtained from the proposed annotation algorithm. For this purpose, we collected additional 10 sequences of data, where the objects in these sequences were equipped with MoCap optical markers, allowing us to obtain ground truth with millimeter-level accuracy. We use the Absolute Trajectory Error (ATE) and Relative Pose Error (RPE) [52] to evaluate the pose annotation compared to the ground truth object pose. The results show that the average rotation error between our poses and those obtained from the MoCap system is less than 0.5 degree, and the average translation error is less than 0.5 cm, which is sufficiently accurate. Due to space limitations, please see the PDF supplementary material on the project page for details of this experiment.

B. Overview

As discussed in section II-A1, IOPE methods are not our primary focus, instead, we adopt the COPE and UOPE methods for benchmarking, which are capable of generalizing across intra-class variations and unseen objects without the need for individual retraining for each object. These methods can directly leverage open-source pre-trained weights. This approach serves two purposes: first, it allows us to verify the validity and accuracy of our data; second, it helps evaluate the generalization capability of the open-source algorithms on our dataset. Furthermore, we fine-tuned several variants of the COPE method on the COPE-119 training set and evaluated

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

TABLE II
COMPARISON OF COPE METHODS ON THE COPE-119 DATASET. METHODS ARE CATEGORIZED INTO THOSE THAT PREDICT NOCS MAPS AND THOSE THAT DIRECTLY PREDICT OBJECT POSES. * INDICATES THAT ONLY THE NOCS MAP PREDICTION COMPONENT WAS USED. † DENOTES THAT THE METHOD DOES NOT PREDICT SCALE.

Method	IoU25 ↑	IoU50 ↑	IoU75 ↑	5°2cm ↑	5°5cm ↑	10°2cm ↑	10°5cm ↑
NOCS [9]	73.50	48.75	20.47	11.57	16.71	22.43	32.98
NOCS (MV-ROPE)* [51]	88.74	71.67	38.19	28.84	31.94	48.50	55.53
DiffusionNOCS [32]	83.00	38.89	4.57	39.87	43.83	55.82	64.80
GCASP [38]	89.13	52.84	14.79	50.22	52.74	65.89	70.97
VI-Net [40]	95.66	87.34	52.35	55.09	59.98	72.35	79.70
IST-Net [35]	97.52	90.68	58.99	51.57	52.85	70.90	73.41
GenPose† [39]	-	-	-	47.48	54.30	60.96	71.84
SecondPose [41]	**98.55**	90.90	53.83	58.57	**64.00**	75.13	**83.29**
AGPose [42]	98.47	**92.94**	**66.34**	**61.61**	63.24	**80.02**	82.06

TABLE III
COMPARISON OF FINE-TUNED VS. ORIGINAL COPE METHODS ON COPE-119 TEST SPLIT, * INDICATES FINE-TUNED METHODS.

Method	IoU25 ↑	IoU50 ↑	IoU75 ↑	5°2cm ↑	5°5cm ↑	10°2cm ↑	10°5cm ↑
SecondPose [41]	98.53	92.24	57.42	61.88	67.05	78.24	84.08
AGPose [42]	98.52	92.10	68.53	67.11	67.50	81.23	81.92
SecondPose* [41]	99.38(+0.85)	96.23(+3.99)	59.80(+2.38)	**76.33**(+14.45)	**82.91**(+15.86)	89.43(+11.19)	**97.19**(+13.11)
AGPose* [42]	**99.91**(+1.39)	**99.76**(+7.66)	**84.42**(+15.89)	75.10(+8.99)	75.74(+8.24)	**94.35**(+13.12)	96.46(+14.54)

their performance on the test set to validate the effectiveness of our ground-truth labels. The fine-tuned models yielded significant performance gains. Additionally, we assessed various object pose tracking methods to further broaden the analysis.

C. Evaluation Metrics

We follow [9] and adopt the evaluation protocol commonly used for COPE methods, which includes the mean precision of 3D intersection over union (IoU) to jointly assess rotation, translation, and size prediction. 3D IoU measures the overlap between the 3D bounding box of an object derived from the predicted pose and the ground truth bounding box, providing a comprehensive metric. For rotation and translation errors, we typically report the accuracy within a specified threshold. For instance, $\text{Rot}_m\text{Trans}_n$ represents the percentage of predicted poses with a rotation error less than m degrees and a translation error less than n centimeters.

We evaluated the Average Recall (AR) [53] for UOPE methods using three error functions: Visible Surface Discrepancy (VSD), Maximum Symmetry-Aware Surface Distance (MSSD), and Maximum Symmetry-Aware Projection Distance (MSPD). In addition, we report the frame rate (FPS) for all UOPE methods.

The evaluation of OPT methods' performance relies on two complementary metrics: We evaluate object pose tracking performance using ATE [52] for global accuracy and RPE [52] for local accuracy, reporting average results across all 175 sequences. We also report the AUC of ADD and ADD-S [12], as well as the average recall of ADD(S) within 10% of the object diameter (termed ADD(S)-0.1d) [54].

D. Category-level Pose Estimation

1) COPE-119 Baseline Evaluation: As shown in table II, for COPE task, we evaluate 9 methods: [9], [32], [51] predict canonical representations directly, while [35], [38], [39], [40], [41], [42] predict object poses. Except for [9], which performs joint object detection and pose estimation, all methods utilize SAM2 [47] segmentation results to ensure a fair benchmark. Notably, [32] achieves a relatively lower IoU score but exhibits a very small pose error. A possible reason is that it has never been trained on real-world data, leading to instability in object scale prediction.

2) COPE-119 Fine-Tuning Evaluation: The purpose of this experiment is to demonstrate the accuracy of pose annotations, based on the assumption that the metric improves only when the ground truth pose annotations in both the training and test sets are correct. We fine-tuned AGPose [42] and SecondPose [41] on our 84-sequence training set for 2 epochs and evaluated them on our 35-sequence test set. As shown in table III, the fine-tuned results significantly outperform their original test set performance by a large margin in merely 2 epochs of fine-tuning, highlighting the accuracy of pose annotations in our dataset.

TABLE IV
COMPARISON OF UOPE METHODS ON THE UOPE-56 DATASETS. * INDICATES POSE ESTIMATION WITHOUT REFINEMENT.

Method	Input	VSD ↑	MSSD ↑	MSPD ↑	AR ↑	FPS ↓
FoundPose* [30]	RGB	50.12	52.02	79.25	60.46	0.50
MegaPose [28]	RGB	64.56	68.45	**83.64**	**72.23**	2.25
GigaPose [29]	RGB	**64.84**	61.32	83.20	69.79	3.37
MegaPose [28]	RGB-D	76.34	80.48	84.64	80.49	2.61
SAM6D [25]	RGB-D	79.67	77.95	81.65	79.75	3.75
FoundationPose [1]	RGB-D	**92.74**	**90.09**	**94.40**	**92.41**	0.98

E. Unseen Object Pose Estimation

We evaluate 5 UOPE methods in table IV, grouped by their input modalities into RGB-based and RGB-D-based approaches. To ensure fairness, each sequence is paired with

a specific CAD model and uniform segmentations produced by SAM2 [47]. Moreover, methods that rely on CNOS [55] for detection or segmentation also utilize SAM2 [47]. Except for [25], which experiences performance bottlenecks due to its intrinsic segmentation component and fixed threshold settings that fail to generalize in our dataset's cluttered backgrounds, we apply dynamic threshold adjustments to maintain accurate pose estimation. [28], available in RGB-only and RGB-D configurations, leverages extensive pre-processing to extract rich template information, demonstrating a significant advantage, especially on single-object sequences with ground-truth detection. Notably, [30] does not open-source its core refinement code, limiting its evaluation of the results, so we only report its coarse results.

F. Object Pose Tracking

For object pose tracking methods, we benchmark [4], [30], [44], [45], [56]. We mainly employed the original algorithms, but with [4], we integrated SAM2-derived masks [47] to zero out weights of non-object regions directly within the DBA layer to derive the camera-to-object poses, effectively adapting it for object pose tracking. We report [1] results under the model-based mode.

TABLE V
COMPARISON ON THE DYNOPETS DATASET. THE EVALUATION OF AVERAGE ATE (M), RPE ON ROTATION (RAD), AND AVERAGE RPE ON TRANSLATION (M).

Method	ATE ↓	RPE Rot. ↓	RPE Trans. ↓
MaskFusion [56]	0.320	0.060	0.033
DROID-SLAM [4]	0.173	0.063	0.040
BundleTrack [44]	0.068	0.037	0.021
BundleSDF [45]	0.094	0.044	0.026
FoundationPose [1]	**0.037**	**0.018**	**0.010**

TABLE VI
COMPARISON ON DYNOPETS DATASET. ADD(S) REPRESENTS AUC (0, 0.1M), AND ADD(S)-0.1D IS THE AVERAGE RECALL WITHIN 10% OF THE OBJECT DIAMETER.

Method	ADD ↑	ADDS ↑	ADD-0.1d ↑	ADDS-0.1d ↑
MaskFusion [56]	12.23	34.47	28.75	64.65
DROID-SLAM [4]	34.27	57.20	62.20	86.41
BundleTrack [44]	76.39	89.36	**94.30**	99.54
BundleSDF [45]	70.64	89.41	92.30	**99.98**
FoundationPose [1]	**89.80**	**96.40**	92.42	99.29

The results are shown in table V and table VI. Due to failure cases in some sequences, [1] underperforms, especially in ADD(S)-0.1d compared to [44] and [45]. Similarly, [56] shows poor performance, consistent with the trajectory error metrics.

V. CONCLUSION

In this work, we propose an efficient framework for acquiring and annotating object pose estimation datasets in environments where both objects and cameras are in motion. Leveraging this approach and our environmental assumptions, we introduce a novel dataset tailored for various object pose tasks. We evaluate 19 state-of-the-art algorithms and provide reference performance benchmarks, validating the high value of the dataset. Future work includes deploying this data acquisition pipeline on a real-world robotic platform to further advance research in robot grasping and manipulation. Another promising direction is to augment the dataset with human hand pose annotations to facilitate research in AR/MR interaction.

VI. ACKNOWLEDGEMENTS

The authors would like to acknowledge the funding support provided by project 62250610225 from the National Natural Science Foundation of China, as well as projects 22DZ1201900, 22ZR1441300, and dfycbj-1 from the Natural Science Foundation of Shanghai. We also sincerely thank Pengfei Tian, Wentao Yang, Yifan Huang and Siao Zhang for their efforts during the data collection process.

REFERENCES

[1] B. Wen, W. Yang, J. Kautz, and S. Birchfield, "Foundationpose: Unified 6d pose estimation and tracking of novel objects," in *Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2024, pp. 17 868–17 879.
[2] S. Ubellacker, A. Ray, J. M. Bern, J. Strader, and L. Carlone, "High-speed aerial grasping using a soft drone with onboard perception," *npj Robotics*, vol. 2, no. 1, p. 5, 2024.
[3] P. Banerjee, S. Shkodrani, P. Moulon, S. Hampali, S. Han, F. Zhang, L. Zhang, J. Fountain, E. Miller, S. Basol, *et al.*, "Hot3d: Hand and object tracking in 3d from egocentric multi-view videos," in *Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2025, pp. 7061–7071.
[4] Z. Teed and J. Deng, "Droid-slam: Deep visual slam for monocular, stereo, and rgb-d cameras," *Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 34, pp. 16 558–16 569, 2021.
[5] E. Olson, "Apriltag: A robust and flexible visual fiducial system," in *IEEE Int. Conf. Robot. Autom. (ICRA)*. IEEE, 2011, pp. 3400–3407.
[6] B. Wen, C. Mitash, B. Ren, and K. E. Bekris, "se (3)-tracknet: Data-driven 6d pose tracking by calibrating image residuals in synthetic domains," in *IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS)*. IEEE, 2020, pp. 10 367–10 373.
[7] B. Drost, M. Ulrich, P. Bergmann, P. Hartinger, and C. Steger, "Introducing mvtec itodd-a dataset for 3d object recognition in industry," in *Int. Conf. Comput. Vis. Workshops (ICCVW)*, 2017, pp. 2200–2208.
[8] A. Guo, B. Wen, J. Yuan, J. Tremblay, S. Tyree, J. Smith, and S. Birchfield, "Handal: A dataset of real-world manipulable object categories with pose annotations, affordances, and reconstructions," in *IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS)*. IEEE, 2023, pp. 11 428–11 435.
[9] H. Wang, S. Sridhar, J. Huang, J. Valentin, S. Song, and L. J. Guibas, "Normalized object coordinate space for category-level 6d object pose and size estimation," in *Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2019, pp. 2642–2651.
[10] M. Zhang, T. Wu, T. Wang, T. Wang, Z. Liu, and D. Lin, "Omni6d: Large-vocabulary 3d object dataset for category-level 6d object pose estimation," in *Eur. Conf. Comput. Vis. (ECCV)*. Springer, 2024, pp. 216–232.
[11] J. Zhang, W. Huang, B. Peng, M. Wu, F. Hu, Z. Chen, B. Zhao, and H. Dong, "Omni6dpose: A benchmark and model for universal 6d object pose estimation and tracking," in *Eur. Conf. Comput. Vis. (ECCV)*. Springer, 2024, pp. 199–216.
[12] Y. Xiang, T. Schmidt, V. Narayanan, and D. Fox, "Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes," in *Robot. Sci. Syst. (RSS)*, 2018.
[13] T. Hodan, P. Haluza, Š. Obdržálek, J. Matas, M. Lourakis, and X. Zabulis, "T-less: An rgb-d dataset for 6d pose estimation of texture-less objects," in *Winter Conf. Appl. Comput. Vis. (WACV)*. IEEE, 2017, pp. 880–888.
[14] S. Hinterstoisser, V. Lepetit, S. Ilic, S. Holzer, G. Bradski, K. Konolige, and N. Navab, "Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes," in *Asian Conf. Comput. Vis. (ACCV)*. Springer, 2012, pp. 548–562.
[15] X. Liu, S. Iwase, and K. M. Kitani, "Stereobj-1m: Large-scale stereo image dataset for 6d object pose estimation," in *Int. Conf. Comput. Vis. (ICCV)*, 2021, pp. 10 870–10 879.

[16] Y. Fu and X. Wang, "Category-level 6d object pose estimation in the wild: A semi-supervised learning approach and a new dataset," *Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 35, pp. 27 469–27 483, 2022.

[17] P. Wang, H. Jung, Y. Li, S. Shen, R. P. Srikanth, L. Garattoni, S. Meier, N. Navab, and B. Busam, "Phocal: A multi-modal dataset for category-level object pose estimation with photometrically challenging objects," in *Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2022, pp. 21 222–21 231.

[18] H. Jung, S.-C. Wu, P. Ruhkamp, G. Zhai, H. Schieber, G. Rizzoli, P. Wang, H. Zhao, L. Garattoni, S. Meier, *et al.*, "Housecat6d-a large-scale multi-modal category level 6d object perception dataset with household objects in realistic scenarios," in *Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2024, pp. 22 498–22 508.

[19] J. Liu, W. Sun, H. Yang, Z. Zeng, C. Liu, J. Zheng, X. Liu, H. Rahmani, N. Sebe, and A. Mian, "Deep learning-based object pose estimation: A comprehensive survey," *arXiv preprint arXiv:2405.07801*, 2024.

[20] C. Wang, D. Xu, Y. Zhu, R. Martín-Martín, C. Lu, L. Fei-Fei, and S. Savarese, "Densefusion: 6d object pose estimation by iterative dense fusion," in *Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2019, pp. 3343–3352.

[21] Z. Dang, L. Wang, Y. Guo, and M. Salzmann, "Learning-based point cloud registration for 6d object pose estimation in the real world," in *Eur. Conf. Comput. Vis. (ECCV)*. Springer, 2022, pp. 19–37.

[22] Y. Xu, K.-Y. Lin, G. Zhang, X. Wang, and H. Li, "Rnnpose: Recurrent 6-dof object pose refinement with robust correspondence field estimation and pose optimization," in *Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2022, pp. 14 880–14 890.

[23] J. Zhou, K. Chen, L. Xu, Q. Dou, and J. Qin, "Deep fusion transformer network with weighted vector-wise keypoints voting for robust 6d object pose estimation," in *Int. Conf. Comput. Vis. (ICCV)*, 2023, pp. 13 967–13 977.

[24] J. Chen, Z. Zhou, M. Sun, R. Zhao, L. Wu, T. Bao, and Z. He, "Zeropose: Cad-prompted zero-shot object 6d pose estimation in cluttered scenes," *IEEE Trans. Circuits Syst. Video Technol.*, 2024.

[25] J. Lin, L. Liu, D. Lu, and K. Jia, "Sam-6d: Segment anything model meets zero-shot 6d object pose estimation," in *Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2024, pp. 27 906–27 916.

[26] A. Caraffa, D. Boscaini, A. Hamza, and F. Poiesi, "Freeze: Training-free zero-shot 6d pose estimation with geometric and vision foundation models," in *Eur. Conf. Comput. Vis. (ECCV)*. Springer, 2024, pp. 414–431.

[27] J. Huang, H. Yu, K.-T. Yu, N. Navab, S. Ilic, and B. Busam, "Matchu: Matching unseen objects for 6d pose estimation from rgb-d images," in *Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2024, pp. 10 095–10 105.

[28] Y. Labbé, L. Manuelli, A. Mousavian, S. Tyree, S. Birchfield, J. Tremblay, J. Carpentier, M. Aubry, D. Fox, and J. Sivic, "Megapose: 6d pose estimation of novel objects via render & compare," in *Conf. Robot Learn. (CoRL)*. PMLR, 2023, pp. 715–725.

[29] V. N. Nguyen, T. Groueix, M. Salzmann, and V. Lepetit, "Gigapose: Fast and robust novel object pose estimation via one correspondence," in *Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2024, pp. 9903–9913.

[30] E. P. Örnek, Y. Labbé, B. Tekin, L. Ma, C. Keskin, C. Forster, and T. Hodan, "Foundpose: Unseen object pose estimation with foundation features," in *Eur. Conf. Comput. Vis. (ECCV)*. Springer, 2024, pp. 163–182.

[31] A. Krishnan, A. Kundu, K.-K. Maninis, J. Hays, and M. Brown, "Omninocs: A unified nocs dataset and model for 3d lifting of 2d objects," in *Eur. Conf. Comput. Vis. (ECCV)*. Springer, 2024, pp. 127–145.

[32] T. Ikeda, S. Zakharov, T. Ko, M. Z. Irshad, R. Lee, K. Liu, R. Ambrus, and K. Nishiwaki, "Diffusionnocs: Managing symmetry and uncertainty in sim2real multi-modal category-level pose estimation," in *IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS)*. IEEE, 2024, pp. 7406–7413.

[33] R. Wang, X. Wang, T. Li, R. Yang, M. Wan, and W. Liu, "Query6dof: Learning sparse queries as implicit shape prior for category-level 6dof pose estimation," in *Int. Conf. Comput. Vis. (ICCV)*, 2023, pp. 14 055–14 064.

[34] Q. Meng, J. Gu, S. Zhu, J. Liao, T. Jin, F. Guo, W. Wang, and W. Song, "Kgnet: Knowledge-guided networks for category-level 6d object pose and size estimation," in *IEEE Int. Conf. Robot. Autom. (ICRA)*. IEEE, 2023, pp. 6102–6108.

[35] J. Liu, Y. Chen, X. Ye, and X. Qi, "Ist-net: Prior-free category-level pose estimation with implicit space transformation," in *Int. Conf. Comput. Vis. (ICCV)*, 2023, pp. 13 978–13 988.

[36] P. Wang, T. Ikeda, R. Lee, and K. Nishiwaki, "Gs-pose: Category-level object pose estimation via geometric and semantic correspondence," in *Eur. Conf. Comput. Vis. (ECCV)*. Springer, 2024, pp. 108–126.

[37] M. Z. Irshad, S. Zakharov, R. Ambrus, T. Kollar, Z. Kira, and A. Gaidon, "Shapo: Implicit representations for multi-object shape, appearance, and pose optimization," in *Eur. Conf. Comput. Vis. (ECCV)*. Springer, 2022, pp. 275–292.

[38] G. Li, Y. Li, Z. Ye, Q. Zhang, T. Kong, Z. Cui, and G. Zhang, "Generative category-level shape and pose estimation with semantic primitives," in *Conf. Robot Learn. (CoRL)*. PMLR, 2023, pp. 1390–1400.

[39] J. Zhang, M. Wu, and H. Dong, "Generative category-level object pose estimation via diffusion models," *Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 36, pp. 54 627–54 644, 2023.

[40] J. Lin, Z. Wei, Y. Zhang, and K. Jia, "Vi-net: Boosting category-level 6d object pose estimation via learning decoupled rotations on the spherical representations," in *Int. Conf. Comput. Vis. (ICCV)*, 2023, pp. 14 001–14 011.

[41] Y. Chen, Y. Di, G. Zhai, F. Manhardt, C. Zhang, R. Zhang, F. Tombari, N. Navab, and B. Busam, "Secondpose: Se (3)-consistent dual-stream feature fusion for category-level pose estimation," in *Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2024, pp. 9959–9969.

[42] X. Lin, W. Yang, Y. Gao, and T. Zhang, "Instance-adaptive and geometric-aware keypoint learning for category-level 6d object pose estimation," in *Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2024, pp. 21 040–21 049.

[43] W. Li, H. Xu, J. Huang, H. Jung, P. K. Yu, N. Navab, and B. Busam, "Gce-pose: Global context enhancement for category-level object pose estimation," in *Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2025, pp. 27 154–27 165.

[44] B. Wen and K. Bekris, "Bundletrack: 6d pose tracking for novel objects without instance or category-level 3d models," in *IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS)*. IEEE, 2021, pp. 8067–8074.

[45] B. Wen, J. Tremblay, V. Blukis, S. Tyree, T. Müller, A. Evans, D. Fox, J. Kautz, and S. Birchfield, "Bundlesdf: Neural 6-dof tracking and 3d reconstruction of unknown objects," in *Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2023, pp. 606–617.

[46] I. Occipital, "Structure sensor," https://structure.io/.

[47] N. Ravi, V. Gabeur, Y.-T. Hu, R. Hu, C. Ryali, T. Ma, H. Khedr, R. Rädle, C. Rolland, L. Gustafson, *et al.*, "Sam 2: Segment anything in images and videos," *arXiv preprint arXiv:2408.00714*, 2024.

[48] G. Bishop, G. Welch, *et al.*, "An introduction to the kalman filter," *Proc. ACM SIGGRAPH Courser*, vol. 8, no. 27599-23175, p. 41, 2001.

[49] H. E. Rauch, F. Tung, and C. T. Striebel, "Maximum likelihood estimates of linear dynamic systems," *AIAA J.*, vol. 3, no. 8, pp. 1445–1450, 1965.

[50] N. Karaev, I. Makarov, J. Wang, N. Neverova, A. Vedaldi, and C. Rupprecht, "Cotracker3: Simpler and better point tracking by pseudo-labelling real videos," *arXiv preprint arXiv:2410.11831*, 2024.

[51] J. Yang, Y. Chen, X. Meng, C. Yan, M. Li, R. Cheng, L. Liu, T. Sun, and L. Kneip, "Mv-rope: Multi-view constraints for robust category-level object pose and size estimation," in *IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS)*. IEEE, 2024, pp. 7588–7595.

[52] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of rgb-d slam systems," in *IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS)*. IEEE, 2012, pp. 573–580.

[53] T. Hodan, M. Sundermeyer, Y. Labbe, V. N. Nguyen, G. Wang, E. Brachmann, B. Drost, V. Lepetit, C. Rother, and J. Matas, "Bop challenge 2023 on detection segmentation and pose estimation of seen and unseen rigid objects," in *Conf. Comput. Vis. Pattern Recog. (CVPR)*, 2024, pp. 5610–5619.

[54] X. He, J. Sun, Y. Wang, D. Huang, H. Bao, and X. Zhou, "Onepose++: Keypoint-free one-shot object pose estimation without cad models," *Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 35, pp. 35 103–35 115, 2022.

[55] V. N. Nguyen, T. Groueix, G. Ponimatkin, V. Lepetit, and T. Hodan, "Cnos: A strong baseline for cad-based novel object segmentation," in *Int. Conf. Comput. Vis. (ICCV)*, 2023, pp. 2134–2140.

[56] M. Runz, M. Buffier, and L. Agapito, "Maskfusion: Real-time recognition, tracking and reconstruction of multiple moving objects," in *IEEE Int. Symp. Mix. Augment. Real. (ISMAR)*. IEEE, 2018, pp. 10–20.