ComPose: When to Trust Hands for Object Pose Tracking

Jisu Shin[1], Junoh Lee[2], JunGyu Lee[3], Inhwan Bae[4], Dohyeon Lee[3], Hokyun Im[3], Youngwoon Lee[5], Hae-Gon Jeon[3]*

[1]GIST [2]Lunit [3]Yonsei Univ. [4]DGIST [5]SNU

{jsshin98}@gm.gist.ac.kr,
{junoh.lee}@lunit.io,
{jungyu.lee, jellyho, earboll}@yonsei.ac.kr,
dohyeon@postech.ac.kr, inhwanbae@dgist.ac.kr, youngwoon@snu.ac.kr

Abstract

Reconstructing the motion of objects from videos is a key component for embodied AI and robot manipulation. While diverse approaches to object pose tracking have been studied, they rely heavily on strong external priors, such as depth data or 3D templates, and remain highly vulnerable to severe occlusions by hand grasps despite the use of explicit masks. In this work, we present ComPose, a 6DoF object tracking framework designed for hand-aware object pose estimation from RGB video. Rather than treating the hand purely as an occluder, our method harmonizes hand motions as a *complementary cue* for object tracking. In detail, we recover a variety of object motions over time by combining object and hand cues from foundation models within a unified tracking pipeline. Here, ComPose adaptively selects informative hand joints, combines object- and hand-derived cues for motion estimation, and refines the resulting object motion using visible geometric evidence and a learned correction. We further enforce the temporal consistency over both rotation and translation, yielding stable 3D object trajectories over time without any robot-specific trajectory optimization. Extensive experiments show that our method is accurate, efficient, and robust under severe hand occlusion and geometric ambiguity. In addition, the resulting trajectories can also effectively transfer to downstream robot manipulation by enabling robots to reconstruct human actions from online videos.

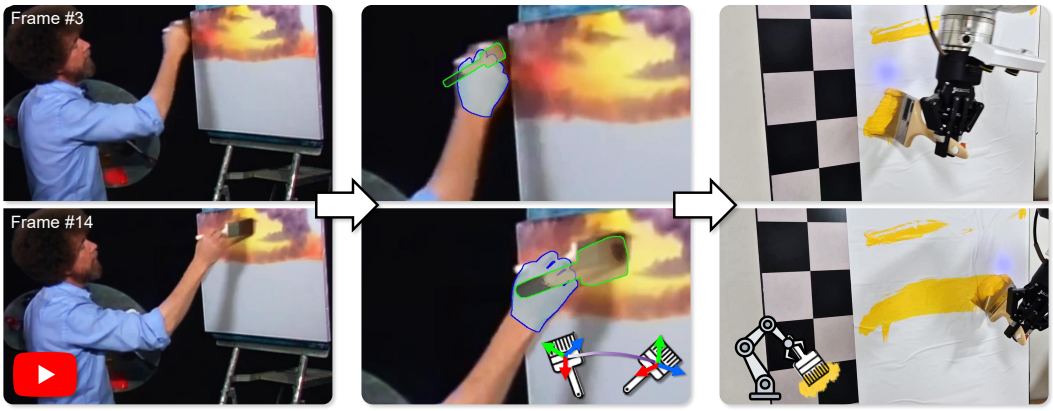
1 Introduction

Object pose estimation and tracking from monocular RGB video is one of the core problems for understanding object motion and dynamic scene interactions in the real-world. Its importance has grown further with recent progress in embodied AI and robotics, where object motions observed in videos can serve as direct guidance for transferring human experts' demonstrations to robotic systems. More broadly, reliable tracking of manipulated objects enables structured understanding of scene dynamics and human-object interaction, making it an important component for video understanding, embodied perception and robot manipulation.

Existing object tracking and pose estimation methods often rely on strong external guidance, such as depth sensors [87, 88, 79], object templates [39, 34], or specialized correspondence modules [74, 57]. In particular, many 6DoF pose estimation methods assume given object-specific shape priors, such as CAD models or ground-truth meshes, which limits their generalizability in real-world settings. Moreover, under hand-object interaction, matching- or correspondence-based methods often become unreliable because large portions of the object are occluded by the hand, leaving only limited visible regions for geometric reasoning. As a result, recovering accurate object motion solely from RGB observations remains particularly challenging in hand-held object tracking scenarios, especially under hand-induced occlusion.

*Corresponding Author

40th Conference on Neural Information Processing Systems (NeurIPS 2026).



(a) YouTube from "*Bob Ross Painting*" (b) Brush Tracking from **ComPose** (c) Our Real-World Robot *Painting*

Figure 1: **Robot manipulation guided by the internet video.** (a) Given an instructional painting video *"Bob Ross Painting"* from Youtube, our method (b) predicts Bob's brush trajectory and (c) the estimated smooth trajectory is directly transferred to a real-world robot for painting.

This implies that reliable hand-held object tracking should not depend on object observations alone. When sufficient object regions are visible, object geometry provides direct evidence for tracking. When the object is heavily occluded by hand, however, hand motion can offer complementary information about how the object moves. The hand trajectory, orientation and grasp configuration physically constrain the motion of the manipulated object, making hand cues strong priors for motion estimation. Motivated by this observation, we incorporate object geometry and hand pose within a unified tracking framework.

In this work, we propose **ComPose**, a **Com**plementary **Pose** tracking framework for hand-held object tracking that adaptively *composes* object- and hand-derived cues. Our method is motivated by a simple question: *when should object tracking trust the hand?* We leverage off-the-shelf 3D foundation models to recover dense 3D geometry without requiring explicit reconstruction pipelines or object-specific priors. However, their outputs remain inherently partial, as they are derived only from visible regions and represented as point clouds rather than complete object geometry. Consequently, geometry-based reasoning becomes unreliable when the object is heavily occluded by the hand. At the same time, hand cues alone are also insufficient, as articulated hand motion does not always correspond to object motion, particularly before stable contact is established or during grasping.

To address this challenge, our adaptive tracking network predicts how object- and hand-derived cues should be used for pose estimation. Specifically, it predicts a gating parameter that controls how much the final rotation should rely on the object cue or the hand cue, and a residual translation parameter that refines the initial translation estimate from visible object geometry. A key challenge in using hand cues is that the articulated hand motion does not always reflect the motion of the manipulated object, moreover, the informativeness of each joint depends on the grasp configuration. We therefore predict which hand joints are most informative for pose estimation, and use them to form a hand-based motion cue that complements the object-based estimate. Notably, these joint weights are learned implicitly from the rotation supervision, without requiring any additional joint-level annotation or auxiliary loss. For rotation, rather than predicting each frame independently, we estimate relative rotations between frames so that the prediction is grounded in frame-to-frame motion relationships. For translation, the residual offset compensates for the bias introduced by partial visibility, since the initial estimate is computed only from the visible object region. Finally, by imposing additional temporal consistency and smoothness constraints, our method produces stable and temporally coherent object trajectories over the full sequence.

Extensive experiments demonstrate that our approach yields accurate and temporally stable 3D object trajectories while remaining efficient in practice. Our method improves robustness under severe hand occlusions and in geometrically ambiguous cases such as symmetric objects, where object-only tracking is often unstable [35, 59]. Furthermore, the resulting trajectories transfer effectively to robotic manipulation without additional robot-specific trajectory optimization, showing

that temporally consistent hand-held object tracking from videos can serve as a practical interface between human demonstrations and robot execution.

Our contributions are summarized as follows:

- We present a framework for hand-held object tracking that enables temporally consistent 3D object trajectory estimation directly from RGB videos, without any external guidance.
- We leverage the hand as a complementary cue under occlusion and propose an adaptive module that selects informative joints and fuses hand and object cues.
- We achieve strong tracking performance and show effective transfer of the predicted smooth trajectories to downstream robot manipulation.

2 Related Work

2.1 6D Object Pose Estimation via Semantic Priors

Pioneering works directly estimate the position and orientation of objects in an RGB image through end-to-end regression [33, 91, 14, 5], 2D detection integrated with dense matching [96, 69] or template matching [32, 68, 54]. To further improve regression performance, various strategies apply iterative refinements through render-and-compare optimizations [39, 34, 93, 29, 46, 35, 59], occlusion handling through segmentation and self-occlusion modeling [26, 13], or leverage synthetic training data and bridge the gap using self-supervised methods with pseudo-labels [19] or with implicit 3D representation rendering [95, 60].

To overcome the inherent instability of direct regression, local feature mapping approaches predict semantic keypoints [62, 63, 3] or 3D bounding box corners [66, 20]. Techniques, which map pixel-wise dense coordinates or discretize object surfaces into fragments [25, 80, 73, 61], have demonstrated significant resilience to severe occlusions. Additionally, notable advancements in robust pose estimation have been made from coordinate-based disentangled networks [40] and multi-representation fusion [70]. Moreover, performance has been further advanced by introducing foundation model prompts [17]. We note that approaches, which use additional depth information from RGB-D images to capture global geometric structures [79, 92, 24, 23, 41, 30, 7, 78, 27, 49], are out-of-scope to our work.

2.2 Generalization to Novel Objects and Video Tracking

Since accurate 3D object models are not always available, a significant line of research focuses on generalizing pose estimation to novel objects without relying on predefined template models like CAD. Pioneering works achieve this by generalizing across intra-class variations using canonical object representations [81]. Subsequent studies have further enabled models to accommodate complex structural deviations within specific categories [75, 6, 90, 43, 44, 28, 38, 45, 99, 9, 98, 57].

Recently, the paradigm has shifted toward fully model-free configurations. To break beyond intra-class boundaries, frameworks directly reconstruct 3D feature correlations from sparse reference images or short video scans [74, 22, 50]. These 3D model-free approaches have naturally extended into the temporal domain for novel-object 6D pose tracking [87, 88]. Furthermore, state-of-the-art methods leverage temporal consistency to jointly optimize object meshes and 6DoF trajectories from RGB video [71], while others achieve fast novel object estimation via single-view retrieval [57]. Most recently, the integration of foundation models [42, 48, 36] has significantly advanced tracking stability in cluttered environments [89, 64].

Hand-based geometric estimation has also explored learned confidence and anatomical cues. HandDGP [77] predicts keypoint confidences for differentiable global hand positioning, while EgoForce [53] incorporates hand and forearm cues into a ray-space solver for hand localization. Our method instead learns how informative each joint's inter-frame motion is for the manipulated object's rigid motion. We use these weights to estimate hand-derived object rotation and adaptively fuse it with object-derived geometry for temporally consistent object tracking.

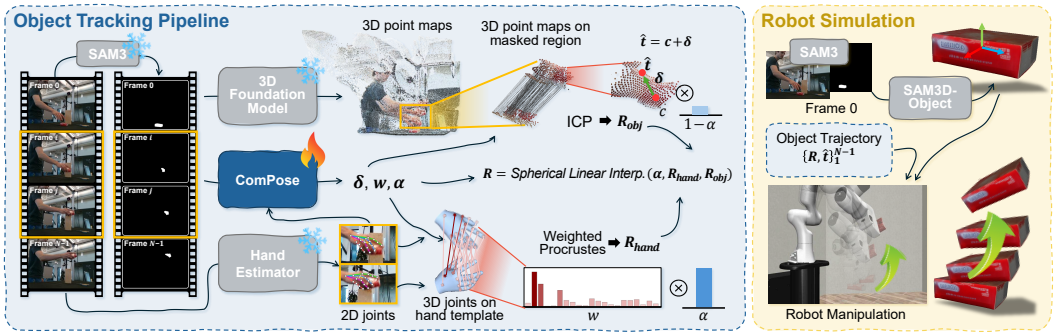


Figure 2: An illustration of the proposed object tracking pipeline and its application to robot simulation.

3 Methodology

Given an input video of hand-object interaction, our goal is to estimate the motion of the manipulated object over time. The proposed framework consists of three components: a 3D foundation model-based tracking, a hand pose estimator-based tracking, and ComPose. Based on dense 3D geometry from the foundation model (Sec. 3.1) and 2D–3D hand joints from the hand estimator, we first obtain object- and hand-based pose cues using the proposed formulation (Sec. 3.2), and then combine them according to their estimated reliability through ComPose (Sec. 3.3).

3.1 Preliminary: 3D Foundation Models

Recent feed-forward visual geometry networks [82, 16, 85, 11, 55] enable dense 3D reconstruction from casual video. Given N input frames $\{I_i\}_{i=1}^{N}$, these models jointly estimate cross-frame consistent dense point maps, depth maps, and camera poses through a unified multi-view transformer f_θ:

$$\{P_i, D_i, C_i\}_{i=1}^{N} = f_\theta\left(\{I_i\}_{i=1}^{N}\right), \tag{1}$$

where $P_i \in \mathbb{R}^{H\times W\times 3}$ is point map, $D_i \in \mathbb{R}^{H\times W}$ refers to depth, and $C_i \in SE(3)$ denotes the camera pose of the i-th frame. In this work, we build on these predictions as the primary geometric representation for object tracking. Based on these dense 3D outputs, our method estimates object pose through our pipeline (Sec. 3.2).

3.2 Pose Tracking

Here, we combine two complementary tracking formulations: a template-free geometric formulation derived from dense 3D object observations, and a hand-based geometric formulation that uses hand motion as an auxiliary cue when object evidence becomes unreliable. These complementary formulations are integrated into our adaptive tracking framework in Sec. 3.3.

Object Template-Free Pose Tracking. Given two frames I_i and I_j, we first obtain per-frame object masks using SAM3 [2]. We then extract dense 3D point maps from a pre-trained 3D foundation model and retain only the object-masked regions to obtain per-frame point clouds $P_i' \in \mathbb{R}^{K_i\times 3}$ and $P_j' \in \mathbb{R}^{K_j\times 3}$, where K_i and K_j denote the number of points reconstructed from the two masked frames. Since explicit point correspondences are unavailable across frames, we estimate the relative object rotation R_{obj} using Iterative Closest Point (ICP) [1]:

$$R_{\text{obj}} = \underset{R\in \text{SO}(3)}{\arg\min} \sum_{k\in K_i} \left\| R(\mathbf{p}_k - \bar{\mathbf{p}}) - (\mathbf{q}_{\pi(k)} - \bar{\mathbf{q}}) \right\|^2, \tag{2}$$

where $\mathbf{p}_k \in P_i'$ denotes a source 3D point, $\mathbf{q}_{\pi(k)} \in P_j'$ is its assigned closest point in the next frame, $\bar{\mathbf{p}}$ and $\bar{\mathbf{q}}$ are the centroids of P_i' and P_j', respectively, and $\pi(k)$ denotes the closest-point assignment updated iteratively during ICP.

In our implementation, we use Flow3R [11] as the 3D foundation model. However, the proposed formulation does not rely on a specific model and can be applied to other 3D foundation models as well, which is validated additional comparisons in Sec. 4.2.

Hand-based Pose Tracking. While the object transformation can be estimated from object point clouds alone, this estimation becomes less reliable when the object is heavily occluded by the hand. To address this issue, our idea is to exploit hand motion as an explicit cue. Since object surfaces do not provide explicit correspondences across frames, whereas hand joints do through the fixed skeletal topology of MANO [67], we estimate object and hand transformations using different alignment strategies. We recover 3D hand joint positions $\mathbf{J}_i, \mathbf{J}_j \in \mathbb{R}^{21\times 3}$ using a hand pose estimator, where the l-th joint in frame i is denoted by $\mathbf{J}_{l,i} \in \mathbb{R}^3$. Because hand joints have known correspondences across frames, we estimate the relative hand rotation R_{hand} using weighted Procrustes alignment [18, 47, 10]:

$$R_{\text{hand}} = VU^\top, \quad \text{where} \quad U\Sigma V^\top = \text{SVD}\left(\sum_{l=1}^{21} w_l \left(\mathbf{J}_{l,i} - \bar{\mathbf{J}}_i\right)\left(\mathbf{J}_{l,j} - \bar{\mathbf{J}}_j\right)^\top\right), \tag{3}$$

where $\bar{\mathbf{J}}_i$ and $\bar{\mathbf{J}}_j$ denote the anchor joint positions (the wrist joint is used in our implementation), and the weights w are predicted by our complementary estimation module described in Sec. 3.3. These weights allow the model to identify which joints are most informative for object motion in a given frame pair, rather than treating all joints equally. In particular, joints involved in object contact or stable grasp formation tend to provide stronger evidence for rigid object motion, whereas other joints may mainly reflect hand articulation that is less relevant to tracking.

In our implementation, we use WiLoR [65] as the default hand pose estimator. However, the proposed formulation is not tied to a specific hand model, which is also addressed in Sec. 4.2.

3.3 ComPose: Complementary Pose Tracking using Object and Hand Cues

Neither object geometry nor hand motion alone provides a consistently reliable cue for rotation estimation in hand-object interaction, as already discussed in Sec. 3.2. Object-based estimation degrades when the visible object region is heavily occluded, whereas hand-based estimation can be inaccurate when articulation or re-grasping happens. To address this, we introduce a complementary pose estimation module that jointly reasons over image observations and hand motion, and predicts both joint-wise confidence weights and a blending coefficient for adaptive hand-object fusion.

Given a temporal window of N frames, we first extract 2D and 3D hand joints using WiLoR [65]. In parallel, we obtain dense image features from DINOv2 [58] and fuse them with the object mask, which are then processed by alternating local self-attention within each frame and global self-attention across frames [82]. Along with this, we extract joint tokens from the 2D hand joint keypoints and update them through cross-attention with the image tokens, followed by global self-attention across all frames. This design allows the model to identify which joints move consistently with the manipulated object and which joints are less reliable due to articulation or unstable contact. It also enables adaptive selection between hand and object cues for each frame pair.

For each temporal pair (i, j) within the sampled window, the module predicts per-joint weights w_{ij} and a blending coefficient $\alpha_{ij} \in [0, 1]$. The predicted weights are used in the weighted Procrustes alignment (Eq. 3) to obtain the hand-based relative rotation estimate $R_{\text{hand},\,ij}$. The blending coefficient determines how much the final rotation should rely on the hand cue. We combine the object- and hand-based relative rotations by spherical linear interpolation in the unit-quaternion space:

$$\hat{q}_{ij} = \frac{\sin\big((1-\alpha_{ij})\theta_{ij}\big)}{\sin\theta_{ij}}\, q_{\text{obj},\,ij} + \frac{\sin\big(\alpha_{ij}\theta_{ij}\big)}{\sin\theta_{ij}}\, q_{\text{hand},\,ij}, \tag{4}$$

$$\theta_{ij} = \arccos\big(|\langle q_{\text{obj},\,ij}, q_{\text{hand},\,ij}\rangle|\big)\,. \tag{5}$$

and convert the interpolated quaternion back to the relative rotation $\hat{R}_{ij}$. This adaptive fusion allows the method to rely more on object geometry when the masked object point cloud is reliable, and to shift toward hand motion when geometric evidence becomes ambiguous.

For translation, we use the centroid of the masked object point cloud as an initial estimate and predict only a residual correction per frame:

$$\hat{\mathbf{t}}_i = \mathbf{c}_i + \boldsymbol{\delta}_i, \qquad \mathbf{c}_i = \frac{1}{|P_i'|}\sum_{\mathbf{p}\in P_i'} \mathbf{p}. \tag{6}$$

where P_i' denotes the object-masked 3D point cloud and $\boldsymbol{\delta}_i$ is predicted from the pooled image feature together with the object centroid $\mathbf{c}_i$. This residual formulation stabilizes translation estimation while compensating for centroid bias caused by partial visibility and hand occlusion.

Training Losses. We train on ordered three-frame windows $a < b < c$, with temporal gaps $b - a$ and $c - b$ sampled independently and uniformly from $\{1, \ldots, 5\}$. We supervise pairwise rotations over $\mathcal{S} = \{(a, b), (b, c), (a, c)\}$ and projected translations in image space, which is more stable under the scale ambiguity of monocular RGB observations, by $\mathcal{L}_{\text{rot}}$ using rotation loss and $\mathcal{L}_{\text{trans}}$ using L1 loss, respectively.

We further impose cumulative consistency $\mathcal{L}_{\text{cons}}$ and temporal smoothness $\mathcal{L}_{\text{smooth}}$:

$$\mathcal{L}_{\text{cons}} = \left\| \log\Big((\hat{R}_{bc}\hat{R}_{ab})^{\top} \hat{R}_{ac} \Big) \right\|_2, \qquad \mathcal{L}_{\text{smooth}} = \left\| \frac{\hat{\mathbf{t}}_c - \hat{\mathbf{t}}_b}{c - b} - \frac{\hat{\mathbf{t}}_b - \hat{\mathbf{t}}_a}{b - a} \right\|_2. \tag{7}$$

where $\hat{R}_{uv}$ denotes the relative rotation estimate for frame pair (u, v). The smoothness term penalizes changes in gap-normalized translation velocities under variable frame sampling.

To prevent the learned translation correction from drifting excessively away from the object point-cloud centroid, we impose a soft bound regularizer $\mathcal{L}_{\text{bound}}$:

$$\mathcal{L}_{\text{bound}} = \left(\max(0, \|\boldsymbol{\delta}_i\| - 0.5\,\sigma_i) \right)^2, \tag{8}$$

where $\boldsymbol{\delta}_i \in \mathbb{R}^3$ is the predicted translation residual for frame i, and σ_i denotes the standard deviation of the object point cloud P_i', used as a scale proxy.

The final objective function $\mathcal{L}$ is defined as follows:

$$\mathcal{L} = \mathcal{L}_{\text{rot}} + \lambda_t \mathcal{L}_{\text{trans}} + \lambda_c \mathcal{L}_{\text{cons}} + \lambda_s \mathcal{L}_{\text{smooth}} + \lambda_b \mathcal{L}_{\text{bound}}, \tag{9}$$

where λ_t, λ_c, λ_s, and λ_b are empirically set to 10.0, 1.0, 1.0, and 1.0, respectively.

4 Experiments

4.1 Experimental Settings

Datasets. We train on DexYCB [4] S3 and HOI4D [51], which together provide diverse view and egocentric hand-object interaction videos with diverse manipulation dynamics. For evaluation, we use the DexYCB S3 test split for in-domain generalization, and HO-3D v2 [21] and OakInk-v1 [94] for cross-dataset evaluation. For OakInk, we exclude bimanual handover and articulated-object sequences to match our single-hand tracking setting.

Metrics. We evaluate object trajectories using both absolute pose accuracy and relative motion accuracy, following trajectory evaluation protocols commonly used in SLAM and visual odometry [72, 100].[2] For relative motion accuracy, we report Relative Translation Error (RTE) and Relative Rotation Error (RRE). These relative metrics quantify how accurately the predicted trajectory captures motion between consecutive frames, thereby reflecting drift and temporal inconsistency. For absolute pose accuracy, we report Absolute Trajectory Error (ATE) for translation and Absolute Rotation Error (ARE). We report RRE and ARE in degrees, and RTE and ATE in the native scale of each dataset.

We additionally report the Temporal Correlation Coefficient (TCC), computed as the Pearson correlation between predicted and ground-truth per-frame motion changes. We compute TCC separately for translation and rotation, and average the results across axes. Let $\log(\Delta R_i), \log(\Delta \hat{R}_i) \in \mathbb{R}^3$ denote the rotation-vector representations of the relative rotations. We define

$$\text{TCC}_{\text{R}} = \frac{1}{3} \sum_{d \in \{x,y,z\}} \rho\Big(\{(\log(\Delta \hat{R}_i))_d\}_{i=1}^{N-1}, \{(\log(\Delta R_i))_d\}_{i=1}^{N-1} \Big), \tag{10}$$

$$\text{TCC}_{\text{T}} = \frac{1}{3} \sum_{d \in \{x,y,z\}} \rho\Big(\{(\Delta \hat{\mathbf{t}}_i)_d\}_{i=1}^{N-1}, \{(\Delta \mathbf{t}_i)_d\}_{i=1}^{N-1} \Big), \tag{11}$$

[2] We report ARE/RRE instead of ADD/AR-style metrics because our setting focuses on video-based object tracking rather than single-frame pose evaluation. These metrics more directly capture trajectory accuracy and temporal consistency for downstream robot manipulation.

Table 1: Main Quantitative Comparison. Time shows the processing time per sequence (Pre-processing / Model Inference Time). † : CAD model used. FoundationPose takes depth estimated by MoGe-2 as an additional input.

Dataset	DexYCB							HO-3D V2							OakInk-V1						
Method	RRE↓	RTE↓	ARE↓	ATE↓	TCC_R↑	TCC_T↑	Time	RRE↓	RTE↓	ARE↓	ATE↓	TCC_R↑	TCC_T↑	Time	RRE↓	RTE↓	ARE↓	ATE↓	TCC_R↑	TCC_T↑	Time
FoundationPose†[89]	3.5	1.7	23.2	4.9	0.28	0.31	(5 / 7)	3.9	0.4	80.1	3.8	0.18	0.11	(64 / 90)	5.7	1.6	30.2	2.9	0.21	0.24	(4 / 4)
MegaPose†[35]	46.1	2.1	62.6	5.0	0.03	0.270	(0 / 610)	16.9	0.5	42.6	2.4	0.02	0.22	(0 / 34K)	64.6	1.8	88.2	4.3	0.01	0.19	(0 / 171)
FoundPose†[59]	55.0	4.9	66.5	6.2	0.02	0.14	(38K / 5)	33.4	1.8	46.0	2.7	0.02	0.10	(39K / 239)	74.7	3.2	92.8	5.6	0.01	0.06	(75K / 32)
FreePose [64]	2.0	0.7	31.2	4.2	0.20	0.51	(0 / 752)	2.6	0.3	80.3	3.4	0.15	0.24	(0 / 5K)	2.7	0.8	30.4	5.1	0.15	0.27	(0 / 1K)
UniHope [86]	44.2	2.1	100.4	12.3	0.01	0.02	(0 / 6)	31.8	0.4	76.9	4.3	0.07	0.04	(0 / 0.2)	-	-	-	-	-	-	-
Ours	1.5	0.6	19.5	3.0	0.55	0.57	(33 / 0.9)	2.4	0.3	49.2	2.7	0.41	0.40	(142 / 9)	2.4	1.2	28.8	3.2	0.19	0.32	(27 / 0.8)

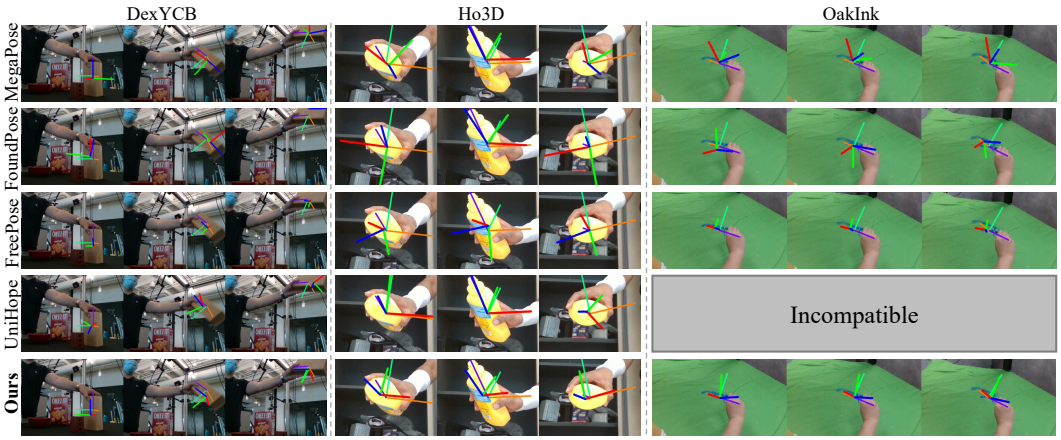


Figure 3: Qualitative results from the state-of-the-art models on three datasets: DexYCB, HO3D-v2, and OakInk-v1. To aid visualization, images are zoomed to highlight the objects.

where $\rho(\cdot,\cdot)$ denotes the Pearson correlation coefficient and $d \in \{x, y, z\}$ indexes the coordinate axes. Complementing error-based metrics, TCC measures temporal motion consistency with the ground truth. Higher values indicate better alignment: 1, 0, and -1 denote perfect positive, no, and perfect negative correlation, respectively.

Implementation Details.

We train our model on an NVIDIA RTX PRO 6000 Blackwell for 30 epochs with a batch size of 128 using AdamW [52], an initial learning rate of 10^{-4}, weight decay of 10^{-4}, and cosine annealing. At inference time, SAM3 [2] provides object masks, while SAM3D-Object [8] estimates the initial object pose and mesh from the first frame for tracking initialization and robot simulation, respectively.

To reduce cumulative rotation drift, we compose relative rotations between anchors spaced $K = 5$ frames apart, including the final frame, into a coarse trajectory. We then correct the frame-by-frame trajectory using anchor discrepancies interpolated via SLERP. For fair evaluation, we align all predictions, including both ours and the baselines, to GT before computing pose errors. For rotation, we align the first frame to remove coordinate-frame ambiguity. For translation, we apply Umeyama alignment [76] over the full trajectory to account for global scale and reference-frame mismatch. More details on datasets, metrics, and implementation are provided in the supplementary material.

4.2 Quantitative and Qualitative Comparisons

We compare our method with both model-based and model-free object pose estimation approaches. As summarized in Tab. 2, prior methods differ in their assumptions on RGB-D input, CAD models, external tracking, and the number of input frames. MegaPose [35] and FoundPose [59] require object CAD models. We also evaluate FoundationPose [89] using CAD models and monocular depth estimated by MoGe-2 [84] instead of

Table 2: Input requirements across models. CoT: Co-Tracker; V: video; * reference-image-based option; ‡ retrieved CAD.

Method	RGB-D	CAD	CoT	Frames
BundleTrack[87]	✓			V
FoundationPose[89]	✓	✓*		1/V
MegaPose[35]		✓		1
FoundPose[59]		✓		1
FreePose[64]		✓‡	✓	V
UniHOPE[86]				1
Ours				V

Table 3: Generalization to egocentric videos on HO-Cap. Camera denotes poses used for world-frame mapping. †: CAD model used.

Method	Camera	RRE↓	RTE↓	ARE↓	ATE↓	TCC_R↑	TCC_T↑
MegaPose†	GT	74.9	1.5	117.0	4.8	0.00	0.09
FoundPose†	GT	95.6	2.7	109.8	5.3	0.00	0.07
FreePose	GT	1.9	0.7	39.0	3.6	0.17	0.25
Ours	GT	1.7	0.6	**31.4**	**3.5**	0.24	0.26
Ours	Flow3R	**1.4**	**0.4**	31.5	4.5	**0.36**	**0.32**

Table 4: Ablation on each module of our pipeline on DexYCB. The default configuration uses Flow3R and WiLoR.

Variant	RRE↓	RTE↓	ARE↓	ATE↓	TCC_R↑	TCC_T↑
Object only ($\alpha = 0$)	1.97	0.60	35.02	3.08	0.37	0.57
Hand only ($\alpha = 1$)	2.64	–	30.65	–	0.38	–
Ours (VGGT)	0.81	0.90	23.72	4.52	0.49	0.47
Ours (HaWoR)	1.52	0.60	19.88	3.08	0.56	0.57
Ours (w/o anchoring)	1.53	0.60	24.48	3.08	0.56	0.57
Ours (GT mask)	1.57	0.43	21.27	2.44	0.55	0.62
Ours (default)	1.5	0.6	19.5	3.0	0.55	0.57

sensor depth, as sensor-depth inputs are beyond the scope of our RGB-only setting. We additionally evaluate BundleTrack [87] using MoGe-2 depth and SAM3 masks, without requiring an object CAD model. These results are provided in Appendix A.3. UniHOPE [86] does not require CAD models, whereas FreePose [64] retrieves a similar CAD model from a large-scale reference database [12, 15] and refines trajectories using CoTracker [31] and PnP [37]. In contrast, our method requires neither CAD models nor retrieved shape priors or reference images of the target object.

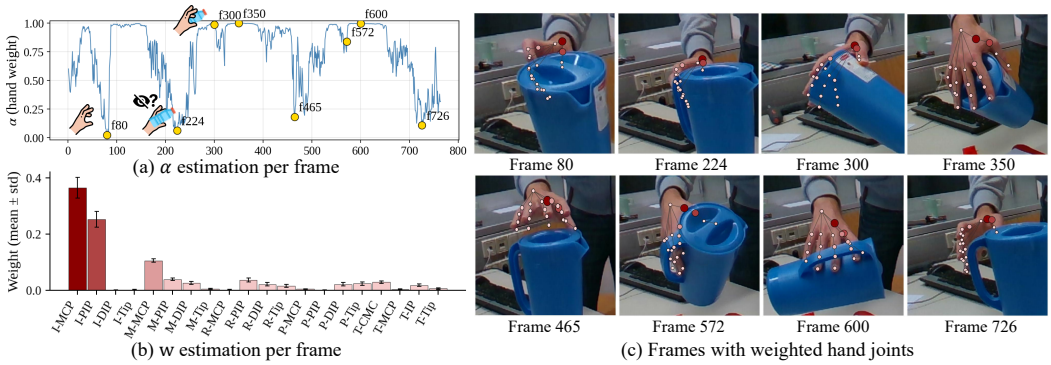
Fig. 3 and Tab. 1 present quantitative and qualitative comparisons with existing approaches. Overall, our method achieves the best or highly competitive performance across most metrics, demonstrating strong temporal consistency in relative object motion. On HO3D, where sequences are substantially longer than in other datasets, absolute metrics such as ARE and ATE are more affected by the accumulated drift when relative predictions are composed over time. Nevertheless, our method remains competitive, showing that the proposed consistency regularization and anchoring effectively suppress cumulative error. On OakInk, we further visualize three consecutive frames to highlight the temporal stability. While the prior methods exhibit noticeable frame-to-frame flickering, our method produces visibly smoother and more stable predictions without any external guidance.

A key advantage of our approach is the inference efficiency. Unlike render-and-match methods such as MegaPose [35] and FoundationPose [89], our tracking module does not require mesh rendering at test time or template matching as in FoundPose [59]. UniHOPE also achieves fast inference, but its object prediction depends on regressing the 2D projections of 3D object corner keypoints and solving PnP, which ties it to object meshes and the object taxonomy seen during training, limiting zero-shot generalization to unseen object classes such as those in OakInk. Methods relying on object appearance or geometric correspondences, including these baselines and FreePose, can further suffer from ambiguity, especially for symmetric, textureless, or heavily occluded objects. In contrast, our method predicts the relative object motion directly from dense 3D geometry and complementary hand cues, making it more robust to symmetry, texture ambiguity and severe hand occlusion, as reflected in both Fig. 3 and Tab. 1.

Generalization to egocentric videos. We evaluate ComPose without retraining on 147 manipulation sequences covering 57 objects from the unseen HO-Cap dataset [83]. For a controlled comparison, we use the same ground-truth camera poses to map all methods' predictions into the world frame. Under this protocol, ComPose outperforms the baselines across all reported metrics (Table 3). Using Flow3R's estimated camera poses also enables tracking without ground-truth camera motion, although camera-pose drift increases world-frame ATE. Sequence selection and evaluation details are provided in Appendix A.1.

Ablation Study. Tab. 4 evaluates the contributions of individual components on DexYCB. Combining object and hand cues improves rotation accuracy over either branch alone, supporting their complementary roles. Anchoring further reduces accumulated rotation errors. Replacing SAM3 masks with ground-truth masks improves translation accuracy and temporal correlation, indicating the importance of accurate visible object regions. Using VGGT yields lower RRE but higher ARE and translation errors than Flow3R, while replacing WiLoR with HaWoR gives comparable performance. These results support our default configuration while showing compatibility with alternative geometry and hand pose estimators.

We then analyze the predicted blending coefficient α and joint weights $\mathbf{w}$ in Fig. 4. As shown in Fig. 4(a), α varies significantly across frames, reflecting the model's adaptive preference between object- and hand-based cues. When the hand is not yet in stable contact or provides weak evidence for object motion, α remains low and the model relies more on object geometry. When the hand



(a) α estimation per frame

(b) w estimation per frame

(c) Frames with weighted hand joints

Figure 4: Analysis on the predicted α and w along with the input frames.

firmly grasps the object and heavily occludes it, α becomes high, indicating greater reliance on the hand cue. Fig. 4(b,c) further shows that higher weights are assigned to joints near the thumb and index contact region, while lower weights are imposed to joints dominated by finger articulation. This behavior supports our weighted Procrustes design, which emphasizes joints that are more reliable for object-aware motion estimation.

We further quantify this behavior in Tab. 5. Hand-object overlap is measured as the proportion of the object mask covered by the hand region. We evaluate overlap levels and manipulation regimes on DexYCB, using HO-Cap separately for re-grasping. As overlap increases, object-only estimates degrade more sharply than hand-only estimates, while our fusion achieves the lowest RRE in each overlap range. Before contact, our fusion matches the object-only estimate and avoids errors from unrelated hand motion. During grasping, it improves over either branch alone. Under strong finger articulation, it performs similarly to the hand-only branch, while re-grasping remains challenging and slightly favors the hand-only estimate.

Table 5: Rotation error (RRE↓) across hand-object overlap levels and manipulation regimes.

Setting	Object only	Hand only	Ours
Hand-object overlap			
0–20%	1.32	1.16	**1.03**
20–40%	2.61	1.55	**1.43**
≥40%	3.45	2.26	**2.01**
Manipulation regime			
Pre-contact	**0.42**	4.42	**0.42**
In-grasping	2.97	1.61	**1.43**
Finger articulation	4.82	3.82	**3.81**
Re-grasping	3.81	**2.30**	2.45

4.3 Application: Robot Manipulation

We directly apply our 6D pose estimates to robotic manipulation. Mapping visual demonstrations to physical robot actions remains challenging due to the need for precise object tracking. Our feed-forward architecture generates temporally consistent and accurate pose sequences, eliminating the need for additional robot-specific post-processing [64]. By streaming predicted poses directly to the controller, our method reduces computational overhead and simplifies real-world deployment.

Direct Trajectory Retargeting. Let $T^{\text{obj}}_{\text{cam}}(t) \in SE(3)$ denote the predicted object pose at time t. With a known camera-to-base transformation $T^{\text{cam}}_{\text{base}}$ and a fixed grasp offset $T^{\text{obj}}_{\text{eef}}$, the target end-effector pose $T^{\text{eef}}_{\text{base}}(t)$ is simply computed as below:

$$T^{\text{eef}}_{\text{base}}(t) = T^{\text{cam}}_{\text{base}} \cdot T^{\text{obj}}_{\text{cam}}(t) \cdot (T^{\text{obj}}_{\text{eef}})^{-1}. \tag{12}$$

This sequence is directly fed into an Operational Space Controller (OSC) driving a Franka Panda robot, avoiding complex optimal control formulations.

Execution in the Real World. We first validate the kinematic stability of these predicted trajectories in the `robosuite` [101] simulation environment before directly deploying them on the physical Franka Panda hardware. We validate this streamlined pipeline by imitating unconstrained *Internet videos* and replaying the motion in the real-world. Figs. 1 and 5 show that the physical robot successfully reproduces fine-grained dynamic motions by directly using the target poses predicted by ComPose. Since we already have camera pose from Flow3R, our method can also directly be applied to egocentric video. More results and further details are provided in supplementary material section and video results.

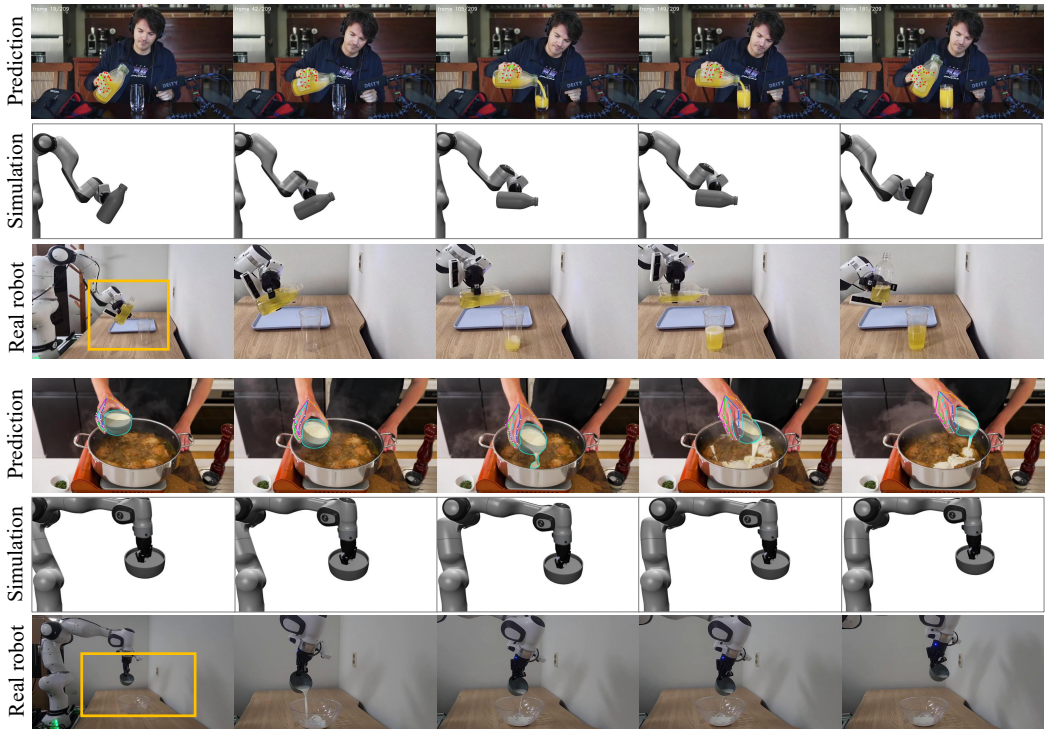


Figure 5: Robot manipulation from internet videos. We overlay the predicted hand joints and mesh of the input frames, and transfer the trajectory to the robot. Our predicted trajectory can be directly utilized without any robot-specific trajectory optimization.

5 Conclusion

We present a novel hand-held object tracking framework that combines object geometry and hand motion under hand-object interaction. By adaptively fusing object- and hand-based cues, our method remains effective in challenging cases where either source alone is insufficient. Extensive experiments across multiple benchmarks demonstrate the effectiveness of the proposed formulation. Moreover, our temporally smooth trajectories can be directly transferred to downstream real-world robot manipulation, enabling stable execution without complex post-hoc optimization.

Limitations. Our method remains challenged when both object geometry and hand motion provide unreliable cues, such as under severe occlusion or motion blur. As shown in Tab. 5, adaptive fusion offers little improvement over hand-only estimation under strong finger articulation and performs slightly worse during re-grasping. These results indicate remaining difficulties when articulated hand motion or changing contact disrupts the correspondence between hand and object motion. Although anchoring reduces rotation error (Tab. 4), it does not eliminate cumulative drift. Future work includes explicit contact-state modeling, training on more diverse interactions, and more robust object and camera trajectory estimation.

Acknowledgments

This work was supported by the Korea Institute of Science and Technology (KIST) Institutional Program (26K0030), the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korean government (MSIT) (RS-2024-00457882), the National Research Foundation of Korea (NRF) grants funded by the Korean government (MSIT) (RS-2024-00338439 and RS-2024-00333634), and the InnoCORE program of the Ministry of Science and ICT (N10250156).

References

[1] Besl, P.J., McKay, N.D.: Method for registration of 3-d shapes. In: Proc. SPIE (1992)

[2] Carion, N., Gustafson, L., Hu, Y.T., Debnath, S., Hu, R., Suris, D., Ryali, C., Alwala, K.V., Khedr, H., Huang, A., Lei, J., Ma, T., Guo, B., Kalla, A., Marks, M., Greer, J., Wang, M., Sun, P., Rädle, R., Afouras, T., Mavroudi, E., Xu, K., Wu, T.H., Zhou, Y., Momeni, L., Hazra, R., Ding, S., Vaze, S., Porcher, F., Li, F., Li, S., Kamath, A., Cheng, H.K., Dollár, P., Ravi, N., Saenko, K., Zhang, P., Feichtenhofer, C.: Sam 3: Segment anything with concepts (2025), https://arxiv.org/abs/2511.16719

[3] Castro, P., Kim, T.K.: Crt-6d: Fast 6d object pose estimation with cascaded refinement transformers. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV) (2023)

[4] Chao, Y.W., Yang, W., Xiang, Y., Molchanov, P., Handa, A., Tremblay, J., Narang, Y.S., Van Wyk, K., Iqbal, U., Birchfield, S., et al.: Dexycb: A benchmark for capturing hand grasping of objects. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2021)

[5] Chen, H., Wang, P., Wang, F., Tian, W., Xiong, L., Li, H.: Epro-pnp: Generalized end-to-end probabilistic perspective-n-points for monocular object pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2022)

[6] Chen, K., Dou, Q.: Sgpa: Structure-guided prior adaptation for category-level 6d object pose estimation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) (2021)

[7] Chen, W., Jia, X., Chang, H.J., Duan, J., Leonardis, A.: G2l-net: Global to local network for real-time 6d pose estimation with embedding vector features. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2020)

[8] Chen, X., Chu, F.J., Gleize, P., Liang, K.J., Sax, A., Tang, H., Wang, W., Guo, M., Hardin, T., Li, X., et al.: Sam 3d: 3dfy anything in images. arXiv preprint arXiv:2511.16624 (2025)

[9] Chen, X., Dong, Z., Song, J., Geiger, A., Hilliges, O.: Category level object pose estimation via neural analysis-by-synthesis. In: Proceedings of the European Conference on Computer Vision (ECCV) (2020)

[10] Choy, C., Dong, W., Koltun, V.: Deep global registration. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2020)

[11] Cong, Z., Zhao, Q., Jeon, M., Tulsiani, S.: Flow3r: Factored flow prediction for scalable visual geometry learning. arXiv preprint (2026)

[12] Deitke, M., Schwenk, D., Salvador, J., Weihs, L., Michel, O., VanderBilt, E., Schmidt, L., Ehsani, K., Kembhavi, A., Farhadi, A.: Objaverse: A universe of annotated 3d objects. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 13142–13153 (2023)

[13] Di, Y., Manhardt, F., Wang, G., Ji, X., Navab, N., Tombari, F.: So-pose: Exploiting self-occlusion for direct 6d pose estimation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) (2021)

[14] Do, T.T., Cai, M., Pham, T., Reid, I.: Deep-6dpose: Recovering 6d object pose from a single rgb image. arXiv preprint (2018)

[15] Downs, L., Francis, A., Koenig, N., Kinman, B., Hickman, R., Reymann, K., McHugh, T.B., Vanhoucke, V.: Google scanned objects: A high-quality dataset of 3d scanned household items. In: 2022 International Conference on Robotics and Automation (ICRA). pp. 2553–2560. Ieee (2022)

[16] Elflein, S., Li, R., Agostinho, S., Gojcic, Z., Leal-Taixé, L., Zhou, Q., Osep, A.: VGG-T^3: Offline feed-forward 3d reconstruction at scale. arXiv preprint (2026)

[17] Fan, Z., Pan, P., Wang, P., Jiang, Y., Xu, D., Wang, Z.: Pope: 6-dof promptable pose estimation of any object in any scene with one reference. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2024)

[18] Gower, J.C.: Generalized procrustes analysis. Psychometrika (1975)

[19] Hai, Y., Song, R., Li, J., Ferstl, D., Hu, Y.: Pseudo flow consistency for self-supervised 6d object pose estimation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) (2023)

[20] Hai, Y., Song, R., Li, J., Salzmann, M., Hu, Y.: Rigidity-aware detection for 6d object pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2023)

[21] Hampali, S., Rad, M., Oberweger, M., Lepetit, V.: Honnotate: A method for 3d annotation of hand and object poses. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2020)

[22] He, X., Sun, J., Wang, Y., Huang, D., Bao, H., Zhou, X.: Onepose++: Keypoint-free one-shot object pose estimation without cad models. Advances in Neural Information Processing Systems (2022)

[23] He, Y., Huang, H., Fan, H., Chen, Q., Sun, J.: Ffb6d: A full flow bidirectional fusion network for 6d pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2021)

[24] He, Y., Sun, W., Huang, H., Liu, J., Fan, H., Sun, J.: Pvn3d: A deep point-wise 3d keypoints voting network for 6dof pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2020)

[25] Hodan, T., Barath, D., Matas, J.: Epos: Estimating 6d pose of objects with symmetries. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2020)

[26] Hu, Y., Hugonot, J., Fua, P., Salzmann, M.: Segmentation-driven 6d object pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2019)

[27] Huang, J., Lin, H., Wang, T., Fu, Y., Xue, X., Zhu, Y.: Cap-net: A unified network for 6d pose and size estimation of categorical articulated parts from a single rgb-d image. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2025)

[28] Irshad, M.Z., Kollar, T., Laskey, M., Stone, K., Kira, Z.: Centersnap: Single-shot multi-object 3d shape reconstruction and categorical 6d pose and size estimation. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA) (2022)

[29] Iwase, S., Liu, X., Khirodkar, R., Yokota, R., Kitani, K.M.: Repose: Fast 6d object pose refinement via deep texture rendering. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) (2021)

[30] Jiang, X., Li, D., Chen, H., Zheng, Y., Zhao, R., Wu, L.: Uni6d: A unified cnn framework without projection breakdown for 6d pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2022)

[31] Karaev, N., Rocco, I., Graham, B., Neverova, N., Vedaldi, A., Rupprecht, C.: Cotracker: It is better to track together. In: European conference on computer vision. pp. 18–35. Springer (2024)

[32] Kehl, W., Manhardt, F.E., Tombari, F., Ilic, S., Navab, N.: Ssd-6d: Making rgb-based 3d detection and 6d pose estimation great again. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) (2017)

[33] Kendall, A., Grimes, M., Cipolla, R.: Posenet: A convolutional network for real-time 6-dof camera relocalization. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) (2015)

[34] Labbé, Y., Carpentier, J., Aubry, M., Sivic, J.: Cosypose: Consistent multi-view multi-object 6d pose estimation. In: Proceedings of the European Conference on Computer Vision (ECCV) (2020)

[35] Labbé, Y., Manuelli, L., Mousavian, A., Tyree, S., Birchfield, S., Tremblay, J., Carpentier, J., Aubry, M., Fox, D., Sivic, J.: Megapose: 6d pose estimation of novel objects via render & compare. In: Proceedings of the Conference on Robot Learning (CoRL) (2022)

[36] Lee, T., Wen, B., Kang, M., Kang, G., Kweon, I.S., Yoon, K.J.: Any6d: Model-free 6d pose estimation of novel objects. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2025)

[37] Lepetit, V., Moreno-Noguer, F., Fua, P.: Ep n p: An accurate o (n) solution to the p n p problem. International journal of computer vision **81**(2), 155–166 (2009)

[38] Li, G., Li, Y., Ye, Z., Zhang, Q., Kong, T., Cui, Z., Zhang, G.: Generative category-level shape and pose estimation with semantic primitives. In: Proceedings of the Conference on Robot Learning (CoRL) (2023)

[39] Li, Y., Wang, G., Ji, X., Xiang, Y., Fox, D.: Deepim: Deep iterative matching for 6d pose estimation. In: Proceedings of the European Conference on Computer Vision (ECCV) (2018)

[40] Li, Z., Wang, G., Ji, X.: Cdpn: Coordinates-based disentangled pose network for real-time rgb-based 6-dof object pose estimation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) (2019)

[41] Li, Z., Stamos, I.: Depth-based 6dof object pose estimation using swin transformer. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (2023)

[42] Lin, J., Liu, L., Lu, D., Jia, K.: Sam-6d: Segment anything model meets zero-shot 6d object pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2024)

[43] Lin, J., Wei, Z., Ding, C., Jia, K.: Category-level 6d object pose and size estimation using self-supervised deep prior deformation networks. In: Proceedings of the European Conference on Computer Vision (ECCV) (2022)

[44] Lin, J., Wei, Z., Li, Z., Xu, S., Jia, K., Li, Y.: Dualposenet: Category-level 6d object pose and size estimation using dual pose network with refined learning of pose consistency. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) (2021)

[45] Lin, Y., Tremblay, J., Tyree, S., Vela, P.A., Birchfield, S.: Single-stage keypoint-based category-level object pose estimation from an rgb image. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA) (2022)

[46] Lipson, L., Teed, Z., Goyal, A., Deng, J.: Coupled iterative refinement for 6d multi-object pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2022)

[47] Lissitz, R.W., Schönemann, P.H., Lingoes, J.C.: A solution to the weighted procrustes problem in which the transformation is in agreement with the loss function. Psychometrika (1976)

[48] Liu, M., Li, S., Chhatkuli, A., Truong, P., Van Gool, L., Tombari, F.: One2any: One-reference 6d pose estimation for any object. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2025)

[49] Liu, X., Wang, G., Li, Y., Ji, X.: Catre: Iterative point clouds alignment for category-level object pose refinement. In: Proceedings of the European Conference on Computer Vision (ECCV) (2022)

[50] Liu, Y., Wen, Y., Peng, S., Lin, C., Long, X.F., Komura, T., Wang, W.: Gen6d: Generalizable model-free 6-dof object pose estimation from rgb images. In: Proceedings of the European Conference on Computer Vision (ECCV) (2022)

[51] Liu, Y., Liu, Y., Jiang, C., Lyu, K., Wan, W., Shen, H., Liang, B., Fu, Z., Wang, H., Yi, L.: Hoi4d: A 4d egocentric dataset for category-level human-object interaction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2022)

[52] Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. arXiv preprint arXiv:1711.05101 (2017)

[53] Millerdurai, C., Wang, S., Xie, Y., Golyanik, V., Stricker, D., Pagani, A.: Egoforce: Forearm-guided camera-space 3d hand pose from a monocular egocentric camera. arXiv preprint arXiv:2605.12498 (2026)

[54] Moon, S., Son, H., Hur, D., Kim, S.: Co-op: Correspondence-based novel object pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2025)

[55] Moreau, A., Shaw, R., Nazarczuk, M., Shin, J., Tanay, T., Zhang, Z., Xu, S., Pérez-Pellitero, E.: Off the grid: Detection of primitives for feed-forward 3d gaussian splatting. arXiv preprint arXiv:2512.15508 (2025)

[56] Mousavian, A., Eppner, C., Fox, D.: 6-dof graspnet: Variational grasp generation for object manipulation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) (2019)

[57] Nguyen, V.N., Groueix, T., Salzmann, M., Lepetit, V.: Gigapose: Fast and robust novel object pose estimation via one correspondence. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2024)

[58] Oquab, M., Darcet, T., Moutakanni, T., et al.: Dinov2: Learning robust visual features without supervision (2024)

[59] Örnek, E.P., Labbé, Y., Tekin, B., Ma, L., Keskin, C., Forster, C., Hodan, T.: Foundpose: Unseen object pose estimation with foundation features. In: Proceedings of the European Conference on Computer Vision (ECCV) (2024)

[60] Park, K., Mousavian, A., Xiang, Y., Fox, D.: Latentfusion: End-to-end differentiable reconstruction and rendering for unseen object pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2020)

[61] Park, K., Patten, T., Vincze, M.: Pix2pose: Pixel-wise coordinate regression of objects for 6d pose estimation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) (2019)

[62] Pavlakos, G., Zhou, X., Chan, A., Derpanis, K.G., Daniilidis, K.: 6-dof object pose from semantic keypoints. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA) (2017)

[63] Peng, S., Liu, Y., Huang, Q., Zhou, X., Bao, H.: Pvnet: Pixel-wise voting network for 6dof pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2019)

[64] Ponimatkin, G., Cífka, M., Souček, T., Fourmy, M., Labbé, Y., Petrik, V., Sivic, J.: 6d object pose tracking in internet videos for robotic manipulation. arXiv preprint (2025)

[65] Potamias, R.A., Zhang, J., Deng, J., Zafeiriou, S.: Wilor: End-to-end 3d hand localization and reconstruction in-the-wild. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2025)

[66] Rad, M., Lepetit, V.: Bb8: A scalable, accurate, robust to partial occlusion method for predicting the 3d poses of challenging objects without using depth. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) (2017)

[67] Romero, J., Tzionas, D., Black, M.J.: Embodied hands: Modeling and capturing hands and bodies together. arXiv preprint (2022)

[68] Shugurov, I., Li, F., Busam, B., Ilic, S.: Osop: A multi-stage one shot object pose estimation framework. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2022)

[69] Shugurov, I., Zakharov, S., Ilic, S.: Dpodv2: Dense correspondence-based 6 dof pose estimation. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) (2021)

[70] Song, C., Song, J., Huang, Q.: Hybridpose: 6d object pose estimation under hybrid representations. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2020)

[71] Song, X., Jin, L., Zhang, Z., Li, J., Zhong, F., Zhang, G., Qin, X.: Prior-free 3d object tracking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2025)

[72] Sturm, J., Engelhard, N., Endres, F., Burgard, W., Cremers, D.: A benchmark for the evaluation of rgb-d slam systems. In: 2012 IEEE/RSJ international conference on intelligent robots and systems. pp. 573–580. IEEE (2012)

[73] Su, Y., Saleh, M., Fetzer, T., Rambach, J., Navab, N., Busam, B., Stricker, D., Tombari, F.: Zebrapose: Coarse to fine surface encoding for 6dof object pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2022)

[74] Sun, J., Wang, Z., Zhang, S., He, X., Zhao, H., Zhang, G., Zhou, X.: Onepose: One-shot object pose estimation without cad models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2022)

[75] Tian, M., Ang Jr, M.H., Lee, G.H.: Shape prior deformation for categorical 6d object pose and size estimation. In: Proceedings of the European Conference on Computer Vision (ECCV) (2020)

[76] Umeyama, S.: Least-squares estimation of transformation parameters between two point patterns. IEEE Transactions on pattern analysis and machine intelligence **13**(4), 376–380 (2002)

[77] Valassakis, E., Garcia-Hernando, G.: Handdgp: Camera-space hand mesh prediction with differentiable global positioning. In: European Conference on Computer Vision. pp. 479–496. Springer (2024)

[78] Wang, C., Martín-Martín, R., Xu, D., Lv, J., Lu, C., Fei-Fei, L., Savarese, S., Zhu, Y.: 6-pack: Category-level 6d pose tracker with anchor-based keypoints. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA) (2020)

[79] Wang, C., Xu, D., Zhu, Y., Martín-Martín, R., Lu, C., Fei-Fei, L., Savarese, S.: Densefusion: 6d object pose estimation by iterative dense fusion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2019)

[80] Wang, G., Manhardt, F., Tombari, F., Ji, X.: Gdr-net: Geometry-guided direct regression network for monocular 6d object pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2021)

[81] Wang, H., Sridhar, S., Huang, J., Valentin, J., Song, S., Guibas, L.J.: Normalized object coordinate space for category-level 6d object pose and size estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2019)

[82] Wang, J., Chen, M., Karaev, N., Vedaldi, A., Rupprecht, C., Novotny, D.: Vggt: Visual geometry grounded transformer. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2025)

[83] Wang, J., Zhang, Q., Chao, Y.W., Wen, B., Guo, X., Xiang, Y.: Ho-cap: A capture system and dataset for 3d reconstruction and pose tracking of hand-object interaction. Advances in Neural Information Processing Systems **38** (2026)

[84] Wang, R., Xu, S., Dong, Y., Deng, Y., Xiang, J., Lv, Z., Sun, G., Tong, X., Yang, J.: Moge-2: Accurate monocular geometry with metric scale and sharp details. Advances in Neural Information Processing Systems **38**, 35928–35959 (2026)

[85] Wang, Y., Zhou, J., Zhu, H., Chang, W., Zhou, Y., Li, Z., Chen, J., Pang, J., Shen, C., He, T.: π^3: Permutation-equivariant visual geometry learning. arXiv preprint (2025)

[86] Wang, Y., Xu, H., Heng, P.A., Fu, C.W.: Unihope: A unified approach for hand-only and hand-object pose estimation. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 12231–12241 (2025)

[87] Wen, B., Bekris, K.: Bundletrack: 6d pose tracking for novel objects without instance or category-level 3d models. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (2021)

[88] Wen, B., Tremblay, J., Blukis, V., Tyree, S., Müller, T., Evans, A., Fox, D., Kautz, J., Birchfield, S.: Bundlesdf: Neural 6-dof tracking and 3d reconstruction of unknown objects. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2023)

[89] Wen, B., Yang, W., Kautz, J., Birchfield, S.: Foundationpose: Unified 6d pose estimation and tracking of novel objects. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2024)

[90] Weng, Y., Wang, H., Zhou, Q., Qin, Y., Duan, Y., Fan, Q., Chen, B., Su, H., Guibas, L.J.: Captra: Category-level pose tracking for rigid and articulated objects from point clouds. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) (2021)

[91] Xiang, Y., Schmidt, T., Narayanan, V., Fox, D.: Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes. arXiv preprint (2017)

[92] Xu, D., Anguelov, D., Jain, A.: Pointfusion: Deep sensor fusion for 3d bounding box estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2018)

[93] Xu, Y., Lin, K.Y., Zhang, G., Wang, X., Li, H.: Rnnpose: Recurrent 6-dof object pose refinement with robust correspondence field estimation and pose optimization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2022)

[94] Yang, L., Li, K., Zhan, X., Wu, F., Xu, A., Liu, L., Lu, C.: Oakink: A large-scale knowledge repository for understanding hand-object interaction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2022)

[95] Yen-Chen, L., Florence, P., Barron, J.T., Rodriguez, A., Isola, P., Lin, T.Y.: inerf: Inverting neural radiance fields for pose estimation. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (2021)

[96] Zakharov, S., Shugurov, I., Ilic, S.: Dpod: 6d pose object detector and refiner. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) (2019)

[97] Zhang, J., Deng, J., Ma, C., Potamias, R.A.: Hawor: World-space hand motion reconstruction from egocentric videos. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2025)

[98] Zhang, J., Wu, M., Dong, H.: Genpose: Generative category-level object pose estimation via diffusion models. In: Proceedings of the Neural Information Processing Systems (NeurIPS) (2023)

[99] Zhang, R., Di, Y., Lou, Z., Manhardt, F., Tombari, F., Ji, X.: Rbp-pose: Residual bounding box projection for category-level pose estimation. In: Proceedings of the European Conference on Computer Vision (ECCV) (2022)

[100] Zhang, Z., Scaramuzza, D.: A tutorial on quantitative trajectory evaluation for visual (-inertial) odometry. In: 2018 IEEE/RSJ international conference on intelligent robots and systems (IROS). pp. 7244–7251. IEEE (2018)

[101] Zhu, Y., Wong, J., Mandlekar, A., Martín-Martín, R., Joshi, A., Lin, K., Maddukuri, A., Nasiriany, S., Zhu, Y.: robosuite: A modular simulation framework and benchmark for robot learning. arXiv preprint arXiv:2009.12293 (2020)

A Appendix

The appendix includes sections as follows:
• Section A.1: More implementation details of our pipeline.
• Section A.2: Robot manipulation details.
• Section A.4: Additional results.

A.1 More implementation details of our pipeline

Network Details.

Table 6: **Detailed network structure of the complementary pose estimation module.** The input consists of DINOv2 patch features, object masks, 2D/3D hand joints over a temporal window of N frames. Input images are center-cropped and resized to 224×224 ($H, W = 224$). Here, B denotes the batch size, N the number of sampled frames, $T = H'W'$ the number of patch tokens, and $D = 384$ the token dimension. The module outputs pair-wise joint weights $\mathbf{w}$, blending coefficients α, and per-frame translation residuals $\boldsymbol{\delta}$.

IDX	STRUCTURE	INPUT	OUTPUT
		Inputs and Preprocessing	
1-1	DINOv2 feature extraction	Image ($B \times N \times 3 \times H \times W$)	$B \times N \times H' \times W' \times 384$
1-2	Mask fusion	DINO feat. + mask ($B \times N \times H' \times W' \times 384$)	$B \times N \times T \times 384$
1-3	Frame positional encoding	OUT1-2	$B \times N \times T \times 384$
1-4	Joint token initialization	Joint2D ($B \times N \times 21 \times 2$), Joint3D ($B \times N \times 21 \times 3$)	$B \times N \times 21 \times 384$
		Image Encoder	
2-1	FrameAttention	OUT1-3	$B \times N \times T \times 384$
2-2	GlobalAttention	OUT2-1	$B \times N \times T \times 384$
2-3	FrameAttention	OUT2-2	$B \times N \times T \times 384$
2-4	GlobalAttention	OUT2-3	$B \times N \times T \times 384$
2-5	FrameAttention	OUT2-4	$B \times N \times T \times 384$
2-6	GlobalAttention	OUT2-5	$B \times N \times T \times 384$
2-7	FrameAttention	OUT2-6	$B \times N \times T \times 384$
2-8	GlobalAttention	OUT2-7	$B \times N \times T \times 384$
2-9	Mean pooling over patches	OUT2-8	$B \times N \times 384$
		Joint Encoder	
3-1	CrossAttention (joint2D, image)	OUT1-4, OUT2-8	$B \times N \times 21 \times 384$
3-2	Joint GlobalAttention	OUT3-1	$B \times N \times 21 \times 384$
3-3	CrossAttention (joint2D, image)	OUT3-2, OUT2-8	$B \times N \times 21 \times 384$
3-4	Joint GlobalAttention	OUT3-3	$B \times N \times 21 \times 384$
		Prediction Heads	
4-1	Rotation weight head	Pair-averaged joint tokens	$B \times 21$
4-2	Softmax(τ)	OUT4-1	Pair-wise weight $\mathbf{w}_{ij} \in \mathbb{R}^{21}$
4-3	Alpha head	Concatenated frame pools (i, j)	$B \times 1$
4-4	Sigmoid	OUT4-3	Pair-wise $\alpha_{ij} \in [0, 1]$
4-5	Centroid encoder	Centroid $\mathbf{c}$ ($B \times N \times 3$)	$B \times N \times 96$
4-6	Translation head	Frame pool + centroid encoding	$\boldsymbol{\delta} \in B \times N \times 3$
		Geometric Outputs	
5-1	Weighted Procrustes	J_i, J_j, $\mathbf{w}_{ij}$	$R_{\text{hand},ij} \in \mathbb{R}^{3\times3}$
5-2	SLERP fusion	$R_{\text{obj},ij}$, $R_{\text{hand},ij}$, α_{ij}	$\hat{R}_{ij} \in \mathbb{R}^{3\times3}$
5-3	Residual translation	$\mathbf{c}_i + \boldsymbol{\delta}$	$\hat{\mathbf{t}}_i \in \mathbb{R}^3$

Object Mask Extraction. To initialize the semantic identity of target objects, we automatically extract per-frame segmentation masks using SAM3 [2]. We visualize the example results on Fig. 6. Specifically, we leverage its zero-shot grounding capabilities by providing class-specific text prompts for each video sequence. Among multiple candidate instances, we identify the target object by selecting the mask whose centroid is closest to the hand, effectively isolating the interacting object from the background. These masks are then bidirectionally propagated across the sequence to maintain temporal consistency. To ensure data quality and computational efficiency, masks containing fewer than 100 pixels are discarded as noise, and sequences exceeding 300 frames are subjected to uniform temporal subsampling.

Mesh Reconstruction. We reconstruct the initial object mesh using SAM3D-Object [8], initialized from the SAM3 masks described above. As shown in Fig. 7, the model produces reasonable

object meshes across diverse categories, even without object-specific templates. However, when the reconstructed mesh is projected back to the image, we occasionally observe slight misalignment with the input object. This behavior is consistent with public user reports on SAM3D-Object that highlight the difficulty of obtaining perfectly aligned renderings from the reconstructed 3D scene, especially when camera alignment is imperfect [8]. Because our work does not rely on additional depth-aware refinement, we use the reconstructed mesh directly and leave tighter image-space alignment to future work.

Choice of Models for Object and Hand. We use WiLoR [65] as the default hand pose estimator because our method mainly relies on accurate frame-level hand pose cues for pairwise object motion estimation. HaWoR [97] is designed for world-space hand motion reconstruction from egocentric videos and incorporates additional temporal components such as camera trajectory estimation, making it less directly aligned with our setting. As a result, WiLoR provides a simpler and more efficient choice in practice, while our ablation study shows that the overall formulation remains compatible with HaWoR.

For object geometry, we use Flow3R [11] as the default 3D foundation model. Flow3R improves the performance of feed-forward visual geometry prediction with optical-flow-based supervision and is particularly effective on dynamic scenes and moving objects. This property is especially relevant to our setting, since manipulated objects in hand-object interaction are inherently dynamic. At the same time, our framework is not tied to a specific 3D foundation model. As shown in the ablation study Tab. 4, alternative backbones such as VGGT [82] can also be used, while Flow3R provides slightly stronger performance in our experiments.

After merging those cues with ComPose, we further apply a simple temporal smoothing to our predicted trajectories. Specifically, we use sliding-window quaternion averaging for rotations and moving-average filtering for translations.

Dataset Details.

Training datasets. We use the DexYCB [4] S3 training and validation split for supervised training. This split contains 17 objects across 850 sequences. Since each interaction sequence is captured from 8 synchronized camera views, DexYCB provides diverse observations of the same hand-object motion under varying viewpoints. We additionally use HOI4D [51], which contains approximately 2,000 egocentric hand-object interaction sequences and complements DexYCB with more diverse manipulation dynamics and visual appearances.

Test datasets. For in-domain evaluation, we use the DexYCB S3 test split, which contains 3 unseen objects and 150 sequences. This split evaluates generalization to previously unseen grasped objects under the same capture setup and annotation protocol. For cross-dataset evaluation, we use the HO-3D v2 evaluation split [21], which contains 13 sequences across 10 YCB objects. We further evaluate on the official test split of OakInk-v1 [94], which contains 57 sequences across 27 objects after filtering. Since OakInk includes single-hand interactions together with bimanual handover sequences and articulated objects, we exclude the latter cases to match our single-hand tracking setting. This benchmark evaluates generalization to novel object geometries under unconstrained hand manipulation. We additionally evaluate on egocentric videos from HO-Cap [83] without retraining. We divide the test recordings into 147 object-centric manipulation sequences covering 57 objects. Each sequence is a contiguous interval in which the target object is visible, contains at least 60 frames, and exhibits sufficient object motion. For comparison, we use the same ground-truth camera poses to map all methods' predictions into the world frame. We also evaluate our full pipeline using camera poses estimated by Flow3R, without ground-truth camera motion.

Evaluation Metric Details. To evaluate the fidelity of frame-to-frame motion, we report Relative Translation Error (RTE) and Relative Rotation Error (RRE). Let $\Delta\mathbf{t}_i = \mathbf{t}_{i+1} - \mathbf{t}_i,\ \Delta\hat{\mathbf{t}}_i = \hat{\mathbf{t}}_{i+1} - \hat{\mathbf{t}}_i$ and $\Delta R_i = R_i^\top R_{i+1},\ \Delta\hat{R}_i = \hat{R}_i^\top \hat{R}_{i+1}$. We then define

$$\mathrm{RTE} = \frac{1}{N-1}\sum_{i=1}^{N-1} \|\Delta\mathbf{t}_i - \Delta\hat{\mathbf{t}}_i\|_2, \qquad \mathrm{RRE} = \frac{1}{N-1}\sum_{i=1}^{N-1} \arccos\left(\frac{\mathrm{tr}(\Delta R_i^\top \Delta\hat{R}_i) - 1}{2}\right). \tag{13}$$

These metrics measure how accurately the predicted trajectory captures translation and rotation changes between consecutive frames, and are therefore sensitive to temporal drift and motion inconsistency.

For absolute pose accuracy, we additionally report Absolute Trajectory Error (ATE) for translation and Absolute Rotation Error (ARE):

$$\mathrm{ATE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N} \|\mathbf{t}_i - \hat{\mathbf{t}}_i\|_2^2}, \qquad \mathrm{ARE} = \frac{1}{N}\sum_{i=1}^{N} \arccos\left(\frac{\mathrm{tr}(R_i^\top \hat{R}_i) - 1}{2}\right). \tag{14}$$

While ATE and ARE reflect frame-wise absolute pose accuracy, RTE and RRE provide a complementary view of trajectory quality by directly evaluating relative motion across frames.

A.2 Robot Manipulation Details

Metric Scale Recovery. Monocular reconstruction is inherently ambiguous up to a global scale. For robot execution, we use the MANO hand reconstructed by WiLoR as an approximate real-world size reference. At a stable-grasp frame, we estimate the video scale from this hand-size reference and use it to rescale the reconstructed trajectory for execution. Since actual hand size varies across individuals, this procedure provides an approximate rather than exact metric scale. We found this approximation sufficient for the robot tasks demonstrated in our experiments.

Initial Object Pose and Grasp Offset. To execute the retargeted trajectory, the rigid transformation between the robot's end-effector and the object, denoted as the grasp offset $T_{\text{eef}}^{\text{obj}}$, must be defined. Because the space of valid grasp poses in $SE(3)$ is theoretically infinite, selecting an appropriate $T_{\text{eef}}^{\text{obj}}$ is important to ensure the entire trajectory remains within the robot's kinematically feasible workspace. An improper offset can cause the manipulator to reach joint limits or kinematic singularities mid-trajectory.

In our experiments, we manually specified a fixed $T_{\text{eef}}^{\text{obj}}$ for each demonstration. Because the Franka Emika Panda is a kinematically redundant 7-DoF manipulator, it possesses sufficient null-space flexibility to comfortably execute the 6D target trajectories without requiring strict grasp optimization. However, to scale this to a fully automated pipeline, this manual step could be readily replaced. For instance, an off-the-shelf grasp synthesizer (e.g., GraspNet [56]) could sample candidate poses, allowing the system to automatically select a grasp that maximizes the robot's manipulability index across the entire sequence.

Direct Trajectory Execution via OSC. Once the grasp offset $T_{\text{eef}}^{\text{obj}}$ and the camera alignment $T_{\text{base}}^{\text{cam}}$ are anchored, the target end-effector trajectory $T_{\text{base}}^{\text{eef}}(t)$ is fully determined. Unlike prior methods that require computationally expensive trajectory optimization libraries (e.g., `Aligator` or `PROXDDP`) to regularize joint velocities and torques from noisy predictions, our pipeline leverages the inherent temporal smoothness and accuracy of our feed-forward model. The computed $T_{\text{base}}^{\text{eef}}(t)$ is directly streamed to the Franka Panda's Operational Space Controller (OSC) at the video frame rate (e.g., 30 Hz). The OSC computes the necessary joint torques at 1000 Hz to track each commanded pose, achieving stable physical execution without additional robot-specific trajectory optimization.

A.3 Additional quantitative results

Additional Baseline Comparisons We extend the main comparison by evaluating depth-based and mesh-dependent methods with inputs estimated from RGB. For FoundationPose, we use either DepthAnything3 with its metric-depth conversion pipeline or MoGe-2 to estimate depth, and evaluate both ground-truth CAD models and single-view meshes reconstructed by SAM3D-Object. We also provide the same SAM3D-Object reconstructions to MegaPose and FoundPose to evaluate these methods without pre-existing CAD models. For BundleTrack, we use MoGe-2 depth and SAM3 masks. BundleTrack requires no object mesh, so SAM3D-Object reconstruction is not used.

Tab. 7 reports these comparisons on DexYCB, HO-3D v2, and OakInk-v1, with BundleTrack evaluated on DexYCB. Replacing CAD models with reconstructed meshes generally increases rotation errors for FoundationPose. MegaPose and FoundPose also remain sensitive to the quality of the reconstructed meshes, which determine their rendering or template-matching inputs. BundleTrack exhibits larger tracking errors under estimated depth, suggesting sensitivity to geometric errors and temporal inconsistencies in monocular depth predictions. ComPose achieves lower relative and absolute rotation errors across the reported comparisons by combining RGB-derived object geometry with complementary hand motion.

Table 7: Additional baseline comparisons using estimated depth and reconstructed meshes. FP denotes FoundationPose. CAD denotes ground-truth CAD models, while SAM3D denotes meshes reconstructed by SAM3D-Object. BundleTrack uses MoGe-2 depth and SAM3 masks without a mesh.

Method	RRE↓	RTE↓	ARE↓	ATE↓	TCC_R↑	TCC_T↑
DexYCB						
BundleTrack + MoGe-2	4.6	2.3	37.0	6.8	0.19	0.18
FP + DA3 + CAD	4.3	1.6	30.7	5.7	0.21	0.26
FP + DA3 + SAM3D	4.4	1.6	32.6	5.4	0.19	0.24
FP + MoGe-2 + CAD	3.5	1.7	23.2	4.9	0.28	0.31
FP + MoGe-2 + SAM3D	3.9	1.6	27.2	5.0	0.24	0.29
MegaPose + SAM3D	61.0	23.2	80.9	60.5	0.00	0.01
FoundPose + SAM3D	74.9	10.4	83.6	16.5	0.00	0.13
Ours	**1.5**	**0.6**	**19.5**	**3.0**	**0.55**	**0.57**
HO-3D v2						
FP + DA3 + CAD	3.9	0.4	83.7	3.9	0.23	0.10
FP + DA3 + SAM3D	4.9	**0.3**	92.8	4.5	0.08	0.04
FP + MoGe-2 + CAD	3.9	0.4	80.1	3.8	0.18	0.11
FP + MoGe-2 + SAM3D	5.4	**0.3**	100.4	4.3	0.09	0.03
MegaPose + SAM3D	35.6	2.4	65.1	10.2	0.00	0.14
FoundPose + SAM3D	61.0	7.1	78.7	13.2	0.00	0.13
Ours	**2.4**	**0.3**	**49.2**	**2.7**	**0.41**	**0.40**
OakInk-v1						
FP + DA3 + CAD	5.8	1.4	35.0	**2.7**	0.15	0.27
FP + DA3 + SAM3D	6.2	1.4	43.2	2.8	0.14	0.21
FP + MoGe-2 + CAD	5.7	1.6	30.2	2.9	**0.21**	0.24
FP + MoGe-2 + SAM3D	6.9	1.9	45.6	3.3	0.09	0.19
MegaPose + SAM3D	46.1	7.2	60.8	20.3	0.07	0.28
FoundPose + SAM3D	67.9	18.3	83.9	27.6	0.02	0.10
Ours	**2.4**	**1.2**	**28.8**	3.2	0.19	**0.32**

Effect of Temporal Filtering Our tracking pipeline applies sliding-window quaternion averaging to rotations and moving-average filtering to translations. These operations reduce local inconsistencies between predictions from overlapping windows. The filtering configuration is fixed across datasets and is not tuned specifically for robot execution.

To isolate its effect, we evaluate ComPose with both filtering operations disabled. Tab. 8 compares this raw output with the full pipeline on all three datasets. We also report FreePose before and after its tracking stage. Without filtering, ComPose maintains lower ARE and ATE than FreePose before tracking across all three datasets. Filtering further improves several metrics, particularly on DexYCB, although its benefits are not uniform across datasets and metrics. These results show that the accuracy of ComPose is not solely attributable to post-processing, while also clarifying the contribution of filtering to the reported results.

Long-Horizon Tracking

We evaluate long-horizon tracking on the eight HO-3D v2 test sequences containing at least 700 frames. For each sequence length N, we evaluate the first N frames of the same sequences. We report ARE at $N \in \{100, 200, \ldots, 700\}$ and ATE at $N \in \{100, 300, 500, 700\}$.

Tab. 9 compares rotation errors with and without anchoring, together with baseline methods. Anchoring reduces ARE at every evaluated length and limits its growth as sequences become longer. Without anchoring, rotation errors increase more strongly because relative rotations are composed frame by frame. Our full model achieves the lowest ARE among the compared methods at each evaluated length, although residual drift remains.

Tab. 10 reports translation errors for ComPose and FreePose. ComPose maintains lower ATE throughout the evaluated range, while both methods show a similar increase with sequence length.

Table 8: Effect of temporal filtering. Ours w/o filtering disables quaternion averaging and translation moving-average filtering. FreePose is evaluated before and after its tracking stage.

Method	RRE↓	RTE↓	ARE↓	ATE↓	TCC_R↑	TCC_T↑
DexYCB						
FreePose w/o tracking	5.0	1.4	40.2	4.9	0.10	0.31
FreePose full	2.0	0.7	31.2	4.2	0.20	0.51
Ours w/o filtering	2.0	0.8	20.1	3.7	0.47	0.48
Ours full	**1.5**	**0.6**	**19.5**	**3.0**	**0.55**	**0.57**
HO-3D v2						
FreePose w/o tracking	5.5	0.5	89.4	3.4	0.08	0.16
FreePose full	2.6	**0.3**	80.3	3.4	0.15	0.24
Ours w/o filtering	2.5	0.6	49.9	**2.7**	0.40	0.35
Ours full	**2.4**	**0.3**	**49.2**	**2.7**	**0.41**	**0.40**
OakInk-v1						
FreePose w/o tracking	5.1	1.3	37.4	5.4	0.05	0.18
FreePose full	2.7	**0.8**	30.4	5.1	0.15	0.27
Ours w/o filtering	2.6	1.3	**28.0**	**3.0**	0.17	**0.35**
Ours full	**2.4**	1.2	28.8	3.2	**0.19**	0.32

Table 9: ARE↓ as a function of sequence length on eight HO-3D v2 sequences containing at least 700 frames. Each column evaluates the first N frames of the same sequences.

Method	$N = 100$	$N = 200$	$N = 300$	$N = 400$	$N = 500$	$N = 600$	$N = 700$
MegaPose	38.6	35.3	32.2	34.6	34.3	35.9	36.7
FoundPose	63.6	57.5	54.6	56.8	56.5	56.9	56.4
FreePose	36.2	50.8	49.6	50.9	52.9	57.5	61.6
Ours w/o anchoring	20.9	25.0	28.0	32.8	33.8	33.8	34.3
Ours	**17.1**	**22.0**	**23.3**	**25.5**	**26.3**	**27.0**	**27.2**

Unlike rotation, ComPose estimates translation directly from the object-point-cloud centroid and a learned residual rather than composing relative translations. Its translation errors therefore do not arise from sequential translation composition, although they remain sensitive to the quality of the underlying geometry and camera estimates. The observed increase in ATE may reflect changes in scene difficulty over longer sequence prefixes.

Object Mask Extraction

We evaluate the object-mask extraction pipeline on the DexYCB test split using mean intersection-over-union (mIoU) against ground-truth object masks. Tab. 11 compares geometry-only point segmentation, SAM3 with class prompts, and the successive addition of hand-proximity instance selection and bidirectional mask propagation.

Class prompts improve mask quality over geometry-only point segmentation, which can have difficulty separating nearby hand and object regions during contact. Selecting the candidate instance closest to the hand further improves accuracy by identifying the manipulated object. Bidirectional propagation yields the highest mIoU by maintaining the selected object's masks across the sequence. These results support the combined use of semantic prompts, hand proximity, and temporal propagation.

A.4 Additional results

We provide results of Flow3R and WiLoR on our test datasets in Fig. 8, and 9. We visualize additional demonstration by transferring trajectory from internet video to robot manipulation. To better match the intended manipulation setup, we conduct the experiment using a small bowl filled with liquid. As shown by the pouring results (Fig. 5), the transferred trajectory is sufficiently stable to handle not only rigid object motion but also liquid manipulation, which is particularly sensitive to trajectory noise and instability.

Table 10: ATE↓ in centimeters as a function of sequence length on the same eight HO-3D v2 sequences.

Method	$N = 100$	$N = 300$	$N = 500$	$N = 700$
FreePose	1.81	2.38	2.78	3.05
Ours	**1.42**	**1.93**	**2.35**	**2.63**

Table 11: Object-mask extraction quality on the DexYCB test split, measured by mIoU against ground-truth masks. The last two rows cumulatively add components to SAM3 with class prompts.

Mask extraction variant	mIoU↑
Geometry-only point segmentation	0.71
SAM3 with class prompts	0.76
+ Hand-proximity instance selection	0.84
+ Bidirectional propagation	**0.92**

Finally, we provide additional qualitative results together with comparisons against prior methods on the benchmark test sets in the attached video. For fair comparison, we additionally implement the trajectory optimization procedure used in the original FreePose paper [64] and apply it in the FreePose robot simulation setting. Even without such robot-specific trajectory optimization, our method produces trajectories that transfer effectively to the real robot, further demonstrating the stability and practical usability of the proposed tracking framework.

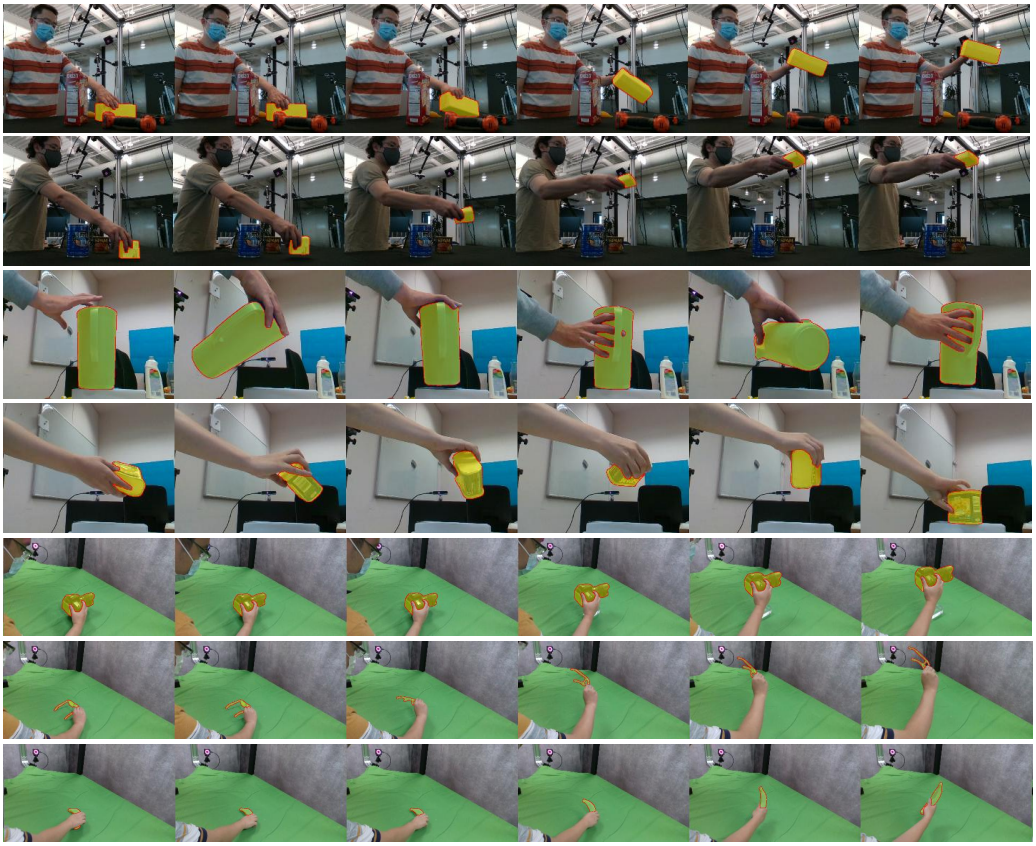


Figure 6: Visualization of mask tracking using SAM3.

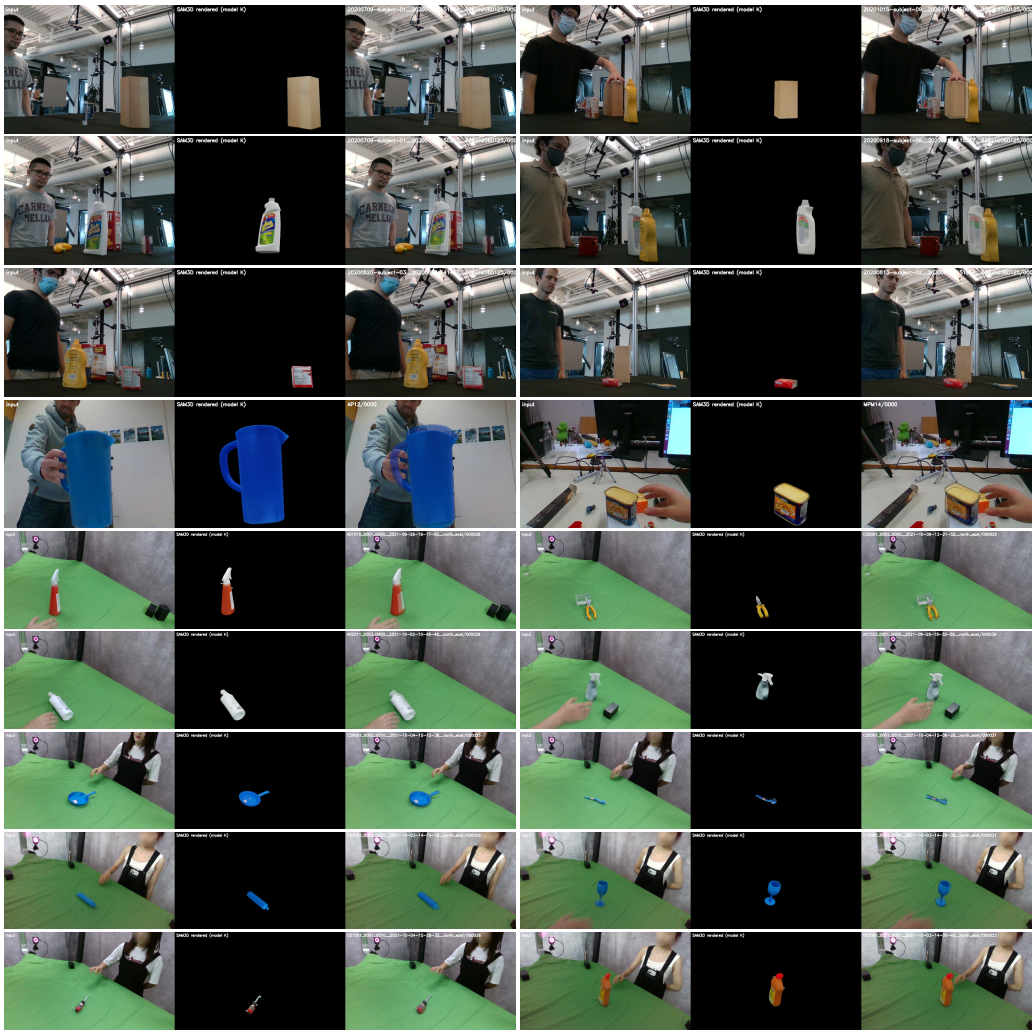


Figure 7: Visualization of mesh reconstruction and pose initialization using SAM3D.

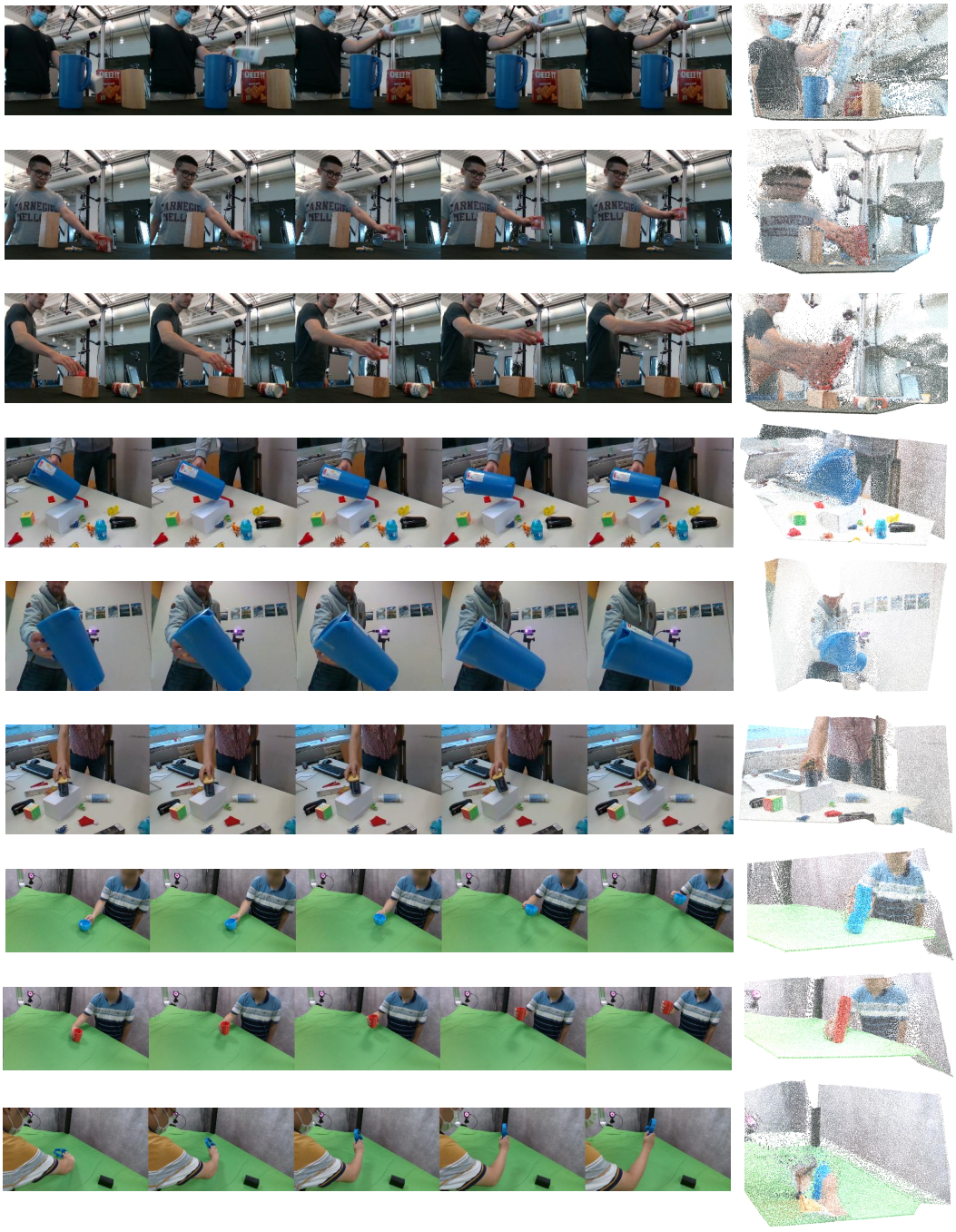


Figure 8: Visualization of point map using Flow3R.

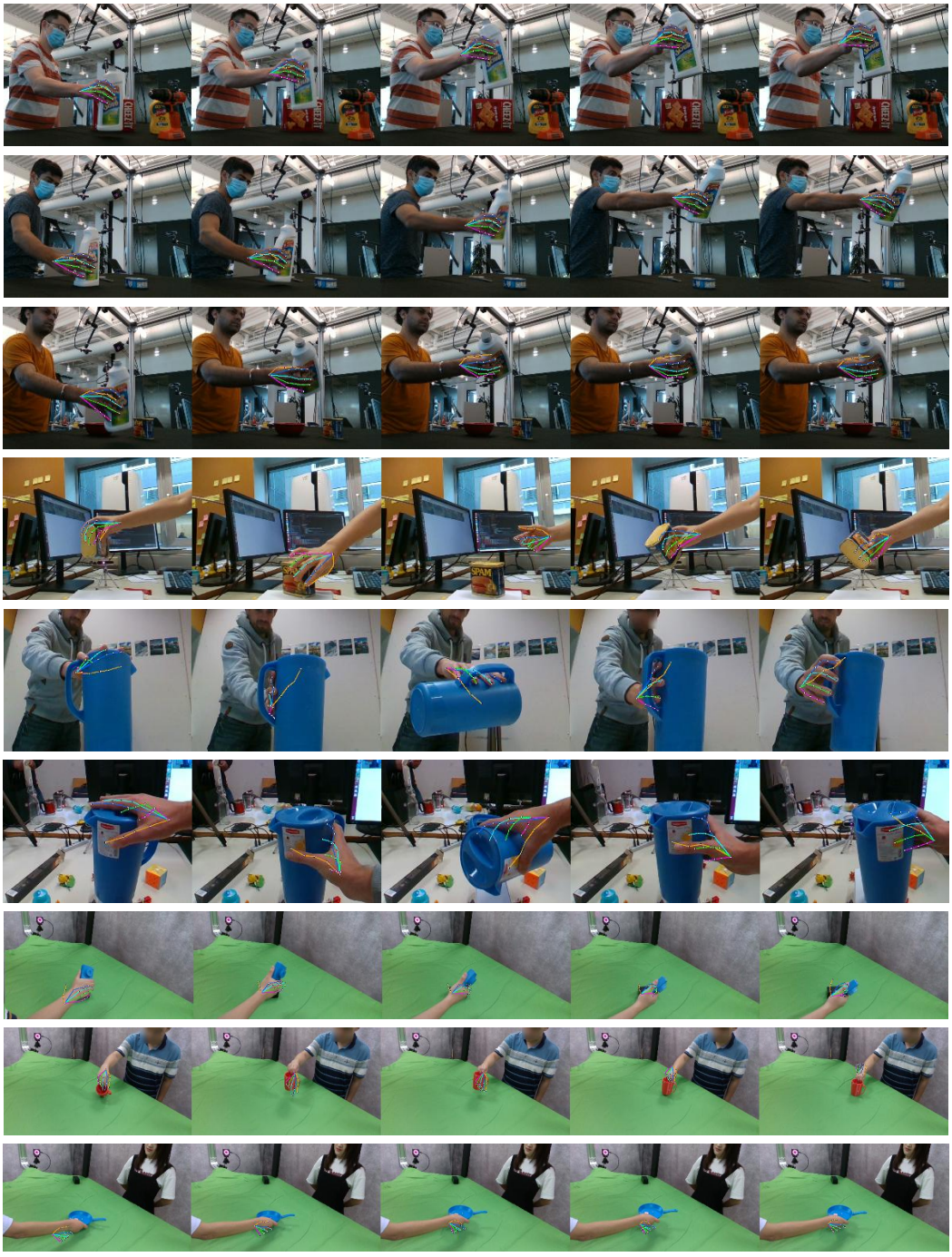


Figure 9: Visualization of 2D hand joints using WiLoR.