Instance Tracking in 3D Scenes from Egocentric Videos

Yunhan Zhao[1] Haoyu Ma[1] Shu Kong[2,3,4] Charless Fowlkes[1]
[1]UC Irvine [2]Texas A&M University [3]Institute of Collaborative Innovation [4]University of Macau
{yunhaz5, haoyum3, fowlkes}@ics.uci.edu skong@um.edu.mo

Abstract

Egocentric sensors such as AR/VR devices capture human-object interactions and offer the potential to provide task-assistance by recalling 3D locations of objects of interest in the surrounding environment. This capability requires instance tracking in real-world 3D scenes from egocentric videos (IT3DEgo). We explore this problem by first introducing a new benchmark dataset, consisting of RGB and depth videos, per-frame camera pose, and instance-level annotations in both 2D camera and 3D world coordinates. We present an evaluation protocol which evaluates tracking performance in 3D coordinates with two settings for enrolling instances to track: (1) single-view online enrollment where an instance is specified on-the-fly based on the human wearer's interactions. and (2) multi-view pre-enrollment where images of an instance to be tracked are stored in memory ahead of time. To address IT3DEgo, we first repurpose methods from relevant areas, e.g., single object tracking (SOT) — running SOT methods to track instances in 2D frames and lifting them to 3D using camera pose and depth. We also present a simple method that leverages pretrained segmentation and detection models to generate proposals from RGB frames and match proposals with enrolled instance images. Our experiments show that our method (with no finetuning) significantly outperforms SOT-based approaches in the egocentric setting. We conclude by arguing that the problem of egocentric instance tracking is made easier by leveraging camera pose and using a 3D allocentric (world) coordinate representation. Dataset and open-source code: https://github.com/IT3DEgo/IT3DEgo.

1. Introduction

Egocentric video obtained from AR/VR devices provides a unique perspective that captures the interaction between the human wearer and the surrounding 3D environment. With the rapid development of AR/VR hardware, there is increasing interest in building assistive agents [41, 57, 59, 68], that track the user's environment and provide contextual guidance on the location of objects of interest (illustrated in Figure 1). We argue that developing such an agent requires solving the largely unexplored problem of *tracking object instances in 3D* from egocentric video.

Why this problem? First, tracking in egocentric video is a novel and underexplored problem, compared to the well-studied tracking from fixed, third-person viewpoints. More broadly, egocentric visual understanding tasks, such as human pose estimation and trajectory prediction [4, 15, 64] are a growing area of interest. Second, tracking in 3D scenes is essential in robotics, autonomous driving, and AR/VR applications. Compared to the 2D counterpart, tracking objects in 3D is crucial for an agent to not only understand the surrounding 3D environment but also to determine precise locations for planning and navigation. Combining the two perspectives above, there is a broader question of what information processing constraints govern how the human visual system integrates egocentric sensory data into a seemingly allocentric perception of the world around us.

Challenges and new opportunities. (1) Egocentric video often features motion blur, hand occlusions, and frequent object disappearances and reappearances which make the 2D tracking problem very challenging from pure visual signals [15, 60]. Tracking in 3D offers an opportunity to fuse additional sensor streams, such as depth and camera pose, to improve accuracy. Unlike 2D tracking with a moving camera, 3D tracking in world coordinates allows the model to leverage the unique prior information – *an object should remain still unless being interacted with the human operator*. (2) For the downstream application of task guidance, we propose exploring novel approaches to identify or *enroll* object instances to be tracked. One approach is automatically enrolling objects with which the user interacts or identifies via hand gestures such as pointing. Alternatively, object instances relevant to a particular task could be *pre-enrolled* based on a collection of images that specify the visual appearance of the object in advance.

Contribution 1: Dataset collection. To our best knowledge, no existing dataset supports exploring the problem of IT3DEgo (c.f. Table 1). The recent Ego4D dataset [23] highlights some of these challenges. However, the Ego4D dataset only provides RGB frames[1] and sparse annotations (may

[1]Ego4D does provide a sparse set of camera poses (less than 15% of frames) estimated with COLMAP and predicted depth maps using monocu-

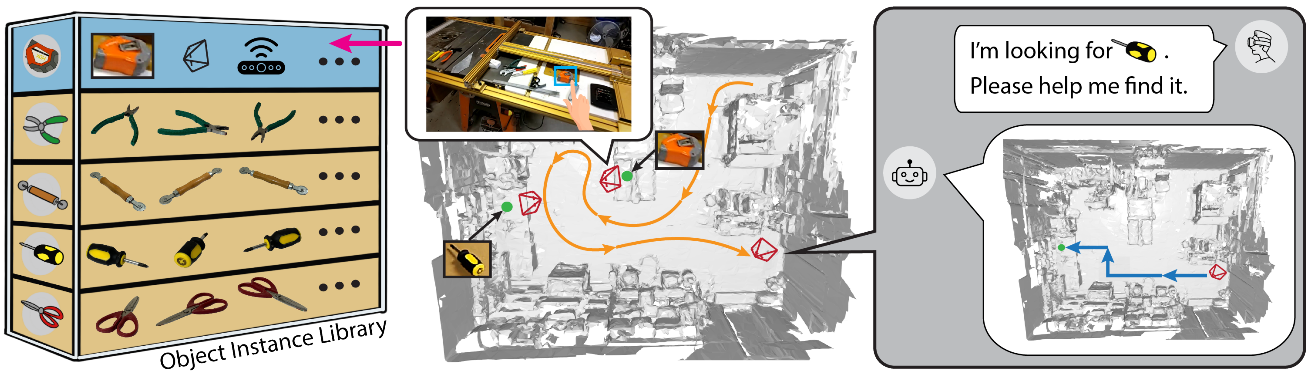

Figure 1. **Motivation for the proposed IT3DEgo benchmark task**. We envision the real-world application of an assistive agent that continuously tracks enrolled object instances in 3D and can provide navigation guidance to users to retrieve object instances at any time. Tracked objects are either enrolled online (first row in the library) where objects of interest are identified automatically based on user interactions or pre-enrolled (bottom four rows in the library), where task-relevant objects are modeled from a collection of photos taken from different views. The former setup comes with additional in-context sensor information, such as camera pose and depth while the latter features richer visual information.

miss potential object location changes), making it unsuitable to fully explore the problem. We collect a new benchmark dataset with HoloLens2, including an RGB camera, a depth sensor, four grayscale cameras, per-frame camera pose and coarse scene geometry as a mesh. We describe the details of dataset statistics, capture procedures, and annotations in Section 3.2.

Contribution 2: Benchmarking protocol. We propose a new IT3DEgo Benchmark for studying instance tracking in 3D scenes from egocentric videos with two settings for how objects are selected for tracking. (1) *Tracking with single-view online enrollment (SVOE)* studies the scenario where object instances of interest are defined on-the-fly, i.e., objects are specified with a 2D bounding box in the frame where they *first* become fully visible to the user. (2) *Tracking with multi-view pre-enrollment (MVPE)* assumes objects of interest are specified by multiple photos of the object of interest from different viewpoints before the tracking system starts. As detailed in Section 3.1, we evaluate performance with standard precision/recall metrics as well as geometric L2 and angular errors used in the Ego4D VQ3D evaluation [23].

Contribution 3: Technical explorations. Since our benchmark task is novel and underexplored in the literature, it is natural to re-purpose and evaluate existing approaches (e.g., SOT methods). We also explore an alternative *piece-wise constant velocity* method that utilizes the Kalman filter [26] with instance proposals from SAM [29] and encoded by DINOv2 [43], resulting in drastic performance improvement over state-of-the-art SOT methods. Section 4 and 5 provide details regarding baselines and benchmark results, respectively. From the experimental results, we provide the following insights: Tracking object instances in egocentric videos is easier in 3D scenes leveraging camera poses and depth maps. Intuitively, an object not being interacted with has the same 3D position in a predefined world coordinate but the positions in 2D frames can change drastically due to the head motion. As a result, existing state-of-the-art 2D SOT approaches perform poorly on egocentric data. Future work should address the problem of re-identifying objects by leveraging the camera poses and accurately identifying and updating object motion changes.

lar depth estimation.

2. Related Work

Egocentric video datasets have been developed to study different problems over the last decade [10, 17, 23, 32, 46, 58]. Traditionally, egocentric video understanding has focused on tasks such as activity recognition [27, 47, 48, 54], human-object interactions [9, 36, 37], and inferring the camera wearer's body pose [25, 40, 52, 64]. Recently, more tasks have emerged due to the increasing interest in egocentric videos, such as action anticipation [18, 19, 51], privacy protection [13, 53, 61], and estimating social interactions [33, 42, 74]. However, object tracking in egocentric videos is largely underexplored in the literature until the introduction of recent datasets[15, 60]. These existing tracking datasets only support 2D tracking, which motivates us to collect and setup a new benchmark to evaluate real-world 3D instance tracking.

Tracking in 3D scenes aims to identify objects of interest in 3D space from a sequence of frames. The prediction output format depends on the downstream tasks, including 3D bounding boxes [28, 67], 3D object centers [73, 79], or 6DOF poses [2, 20]. State-of-the art 3D tracking models [6, 38, 78] have focused on well-established third-person perspective benchmark datasets [5, 8, 21]. The recent large-scale Ego4D dataset starts to address the problem of querying the 3D positions of objects from a first-person perspective.

Table 1. **Comparisons of egocentric datasets that explore tracking-related problem.** Existing egocentric datasets only explore the tracking problem in 2D or predicting discrete 3D locations. Some mention the tracking problem in 3D but only consider limited sensor data (RGB) or synthetic environments. Our benchmark dataset supports the study of instance tracking in 3D real-world scenarios (RWS in the table) from egocentric videos.

Dataset	Modality	Device	Avg. Length	Annot. FPS	RWS	Camera Trajectory	3D Tracking	Year
TREK-150 [14]	RGB	GoPro	10s	60	✓	Natural	✗	2021
EK-VISOR [10]	RGB	GoPro	12s	0.9	✓	Natural	✗	2022
Ego4D-VQ3D [23]	RGB	GoPro	-	-	✓	Natural	✗	2022
EMQA [12]	RGB-D+IMU	-	-	-	✗	Simulated	✗	2022
EgoPAT3D [35]	RGB-D+IMU	Kinect	4min	30	✗	Object-Centric	✗	2022
DigitalTwin [44]	RGB-D+IMU	Aria	2min	-	✗	Natural	✓	2022
EgoTrack [60]	RGB	GoPro	6min	5	✓	Natural	✗	2022
Ours	RGB-D+IMU	HoloLens	>5min	6	✓	Natural	✓	2023

However, the raw sensor data in Ego4D only includes RGB images and no other 3D information, such as depth and camera poses [75, 76]. However, contemporary AR/VR headsets come with additional cameras, depth, and IMU sensors that allow for richer geometric reasoning [44, 62]. Therefore, we believe it is realistic to leverage diverse sensor streams and explore the egocentric tracking problem in 3D. Our benchmark dataset thus includes multiple raw sensors and derived data streams to support the study tracking in 3D scenes with modern hardware platforms.

Object instance detection and tracking is a long-standing problem in computer vision and robotics [16, 22, 24, 56, 63]. Instead of predicting labels from a predefined set of object categories, instance-level predictions treat every object instance as a separate category. Instance-level tracking aims to locate given object instances in a sequence of frames, commonly using a tracking-by-detection paradigm. One common formulation is person re-identification [71, 77], which aims to track and associate individual people as they enter and leave multiple cameras' fields of view. Our setting is closely related but is dominated by the motion of the (egocentric) camera rather than the dynamics of object motion.

3. IT3DEgo: Protocol and Dataset

The problem of IT3DEgo is motivated by real-world assistive agents running on AR/VR devices. Given an object instance specified by the end user, developed models are required to track it in the 3D environment, i.e., recording its 3D location over time (cf. Fig. 2). In this section, we introduce our benchmarking protocol and dataset.

3.1. Benchmarking Protocol

Because object instances of interest are naturally diverse and may fall outside of the vocabulary of existing detectors, we set up a benchmarking protocol that focuses on evaluation without a separate training set. In other words, models should be pretrained on other data sources and cannot see objects in our dataset. This aligns with the contemporary foundation models (e.g., CLIP [49] and SAM [29]) pretrained on open-world data.

Instances enrollment. We consider two distinct setups to specify object instances of interest. The first is *single-view online enrollment (SVOE)*, similar to single object tracking (SOT) where an object is specified on-the-fly by the end users. For example, the user can specify an object of interest by interacting or pointing to it, after which the system should track it in the 3D world. The second is *multi-view pre-enrollment (MVPE)*, which defines (or pre-enrolls) concerned objects with a set of object-centric images captured from multiple angles. The two setups present different challenges. SVOE provides a bounding box of the object (similar to specifying an object in SOT), but the visual quality is generally lower in resolution as the objects can be far from the camera. MVPE provides 25 high-resolution (2124×2832) object-centric images of the instances captured from different angles. However, the object instance is captured under different lighting conditions than the tracking environment, and can be posed differently (e.g., keys can be deformed over time).

Evaluation protocols. Following the literature on object tracking and detection, we use the metrics below in our benchmarking protocol.

- **Precision and recall** at different L2 distance thresholds. Given N specific thresholds τ_i with $i \in \{1, 2, ..N\}$, specifically 0.25, 0.5, 0.75, 1.0, and 1.5 meters, a ground-truth object location $\mathbf{o_{gt}} \in \mathbb{R}^3$ and a predicted location $\mathbf{o_{pred}} \in \mathbb{R}^3$, we count a true positive (TP$_i$) when $||\mathbf{o_{gt}} - \mathbf{o_{pred}}||_2 \leq \tau_i$. At each timestamp, each ground-truth is matched to the prediction with the smallest L2 distance below the threshold. Unmatched predictions and ground-truth at threshold τ_i are counted as false positives (FP$_i$) and false negatives (FN$_i$), respectively. TP$_i$, FP$_i$, and FN$_i$ are computed over all object instances in every frame. The precision and recall at threshold τ_i is computed as $\sum$TP$_i$ / ($\sum$TP$_i$ + $\sum$FP$_i$) and $\sum$TP$_i$ / ($\sum$TP$_i$ + $\sum$FN$_i$), respectively [50, 72].
- **L2 and angular error.** Following VQ3D in Ego4D [23], we also compute the L2 distance between the ground-truth

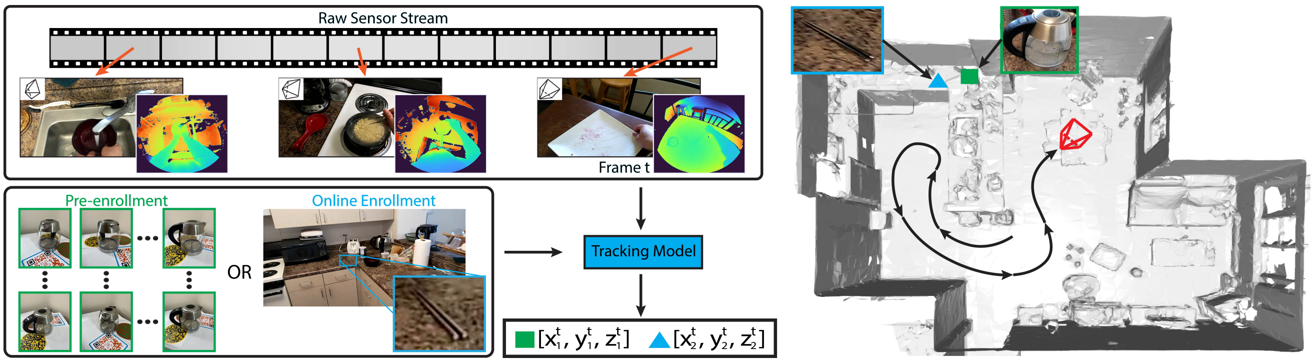

Figure 2. **Illustration of input and output of our benchmark task**. Given a raw RGB-D video sequence with camera poses and object instances of interest, i.e., either by online enrollment (SVOE) or pre-enrollment (MVPE), the goal of our benchmark task is to output the object instance 3D centers in a predefined world coordinate at each timestamp. Please check Section 3.1 for more details.

and predictions in the world coordinates in meters. We also report the angular error in radians in the current camera coordinate system. Unlike threshold-aware 3D precision and recall, these metrics are computed only on frames where both ground-truth and prediction of the object instance location are available.

To make 3D annotation tractable, we only evaluate predictions during time intervals when target objects are stationary (i.e., not being handled by the camera wearer).

3.2. Dataset

Raw video collection. The raw IT3DEgo data was recorded by three individuals in ten diverse indoor scenes, e.g., kitchen, garage, office, labs, etc. The participants perform naturalistic tasks with different object instances in the scene, e.g., cooking, repairing, writing, etc. The raw data includes 50 recordings in total. Each recording contains five or more object instances, each of which appears at three different 3D locations on average. The average length of each recording is 10K frames or >5min. We capture the raw data with HoloLens2 (see Suppl.) which includes an RGB camera, a depth sensor, and four grayscale cameras with the resolution of 720×1280, 480×640, 512×512, respectively. Raw sensors operate at different frequencies, we sync all other sensors to the frequency of the RGB camera (30 fps). We also provide a coarse resolution scene mesh of each environment reconstructed by the Hololense OS. Additional details of the video sequences are described in the supplementary material.

Object instance collection. To support the SVOE setup, annotators identify the first RGB frame where a given object is fully visible and close enough to specify a 2D bounding box which is at least 500 pixels in area. For MVPE, we collected 25 high-resolution images of each object instance using iPhone 13 Pro. Each object was placed on a rotary table with QR codes. We took 12 photos of each object evenly from 360° while keeping the camera at about 30° elevation, 12 more at 60° elevation and 1 top-down view. We provide additional details and visualizations of object instances in the supplementary material.

Annotations. Our dataset includes three types of manual annotations: (1) *Object instance 3D centers* describe the 3D positions of each object instance center in a world coordinate frame. We annotate the 3D center by first averaging 3D points computed from camera poses and depth maps of different views of the object instance. Annotators then examine and adjust computed 3D points by visualizing them together with the coarse mesh of the scene. (2) *2D bounding box annotations* are axis-aligned 2D bounding boxes of the instance every five frames starting from the beginning of the video. Specifically, we ask annotators to draw *amodal* bounding boxes of each object instance. We do not annotate the object instances with heavy occlusions (i.e., when less than 25% of the object is visible). (3) *Object motion state annotations* are a per-frame annotation of whether the object is stationary or dynamic. For the data we collected, dynamic implies the camera wearer is interacting with the object.

4. Methodology

4.1. Baseline: Re-purposed SOT Trackers

To approach the problem of IT3DEgo, we first explore a simple *unified pipeline* as the baseline approach based on single object tracking (SOT). It allows instance-level 2D tracking by providing the visual appearance of object instances to track [3, 11, 70], which enables us to re-purpose them for our benchmark task. In the unified pipeline, we first compute the 2D trajectories of each object instance with SOT. The final 3D trajectories are computed by lifting the center of 2D bounding boxes with depth maps and camera poses. Lastly, we adopt a simple memory mechanism that stores the previous locations of each object instance to handle the case where the instance moves out of sight, i.e., frames without valid predictions.

Lifting 2D trajectories to 3D. With the 2D trajectory predicted from SOT, each valid 2D detection is then lifted into 3D space with the equation: $\mathbf{o}_t^i = \mathbf{T}_t z \mathbf{K}^{-1} \mathbf{c}_t^i$, where $\mathbf{c}_t^i$ is the 2D coordinate of the center of the bounding box of instance i at timestamp t, $\mathbf{o}_t^i$ is the 3D position of instance i at timestamp t in world coordinate. z is the corresponding depth value of $\mathbf{c}_t^i$ on the depth map. $\mathbf{T}_t$ is the camera pose at timestamp t that specifies the camera rotation and translation w.r.t to a predefined world coordinate. $\mathbf{K}$ is the intrinsic matrix. A frame may lack a valid 3D prediction because either there is no 2D location from SOT (e.g., the object is outside the field-of-view) or the depth map is missing the depth value at $\mathbf{c}_t^i$.

Completing 3D trajectories with memory. Any given frame may lack a valid 3D prediction, either because there is no 2D location from SOT (e.g., the object is outside the field-of-view) or the depth map has missing depth values at $\mathbf{c}_t^i$. To address this we implement a simple memory mechanism that stores only the most recent 3D location for each tracked instance (memory size=1). We update the memory whenever there is a new valid prediction. We note that this heuristic is a good match for the prior that object locations change only when they are being interacted with, in which case they should also be visible to the camera.

4.2. Improved Baseline

We also explore the approach that leverages the recent foundation model SAM [29] and state-of-the-art feature encoder DINOv2 [43] for IT3DEgo. Following a tracking-by-detection pipeline, we first compute the per-frame 2D detections of each object instance by comparing the cosine similarity of DINOv2 encoded features between candidate proposals from SAM and a visual feature template. Together with the depth and camera pose information, we convert the 2D detection of each object into a 3D point in a predefined world coordinate. A simple memory with size 1 is also adopted to handle the frames without valid predictions.

Exploring motion prior with Kalman filters. Currently, the naive update mechanism, i.e., always updating the memory for all incoming predictions, does not exploit the temporal information in video sequences. Inspired by the Kalman filter [45, 66, 67] that is widely adopted in the tracking literature, we simply model the stationary position of each object instance as *piecewise constant velocity motion*, leveraging the prior information that an object without being interacted with has the same 3D coordinate. Mathematically, the motion update with Kalman filter in each stationary position: $\hat{\mathbf{x}}_{t+1} = \hat{\mathbf{x}}_t + \mathbf{K}_t(\mathbf{z_t} - \mathbf{H}\hat{\mathbf{x}}_\mathbf{t})$, where $\hat{\mathbf{x}}_t$ is a 6DOF estimated state vector including position and velocity at time step t, $\mathbf{K}_t$ is the Kalman gain, $\mathbf{z_t}$ is a 3DOF the measurement vector, $\mathbf{H}$ is the observation matrix. Please refer to Kalman [26] for more details. Moving from one stationary position to the next one, we introduce an L2 distance heuristic to model the period where objects are being interacted. Specifically, we compute the L2 distance between incoming 3D positions and the state predictions from the Kalman filter. If the L2 distance is above the threshold, we reset the Kalman filter with the current 3D predictions as the initialization.

5. Experiments

In this section, we first describe the implementation details of benchmark results. Then, we show the quantitative results of both setups and the visualizations of tracking results. Lastly, we demonstrate the importance of exploiting camera pose for tracking in 3D and perform ablation studies of the trackers. Note that we split our benchmark dataset into validation and test sets. All experiments are conducted on the validation set; the test set is used for future work.

Baseline SOT trackers. We choose top-ranked trackers from well-established SOT literature and VOT challenges with open-source code for both tracking setups. Specifically, we benchmark three short-term trackers ToMP [39], MixFormer [7], and ARTrack [65]; and three top-performing trackers from VOT long-term tracking challenges 2021 [30] and 2022 [31], mixLT, mlpLT and VITKT_M. We also evaluate trackers that utilize additional depth information as part of the input, including SAMF and MixForRGBD from VOT RGB-D tracking challenge 2022 [31], and ViPT [80]. Lastly, we benchmark the recent egocentric specific finetuned trackers, EgoSTARK [60]. Note that SOT trackers require initial bounding boxes to track, which are not available in MVPE. When re-purposing to MVPE setup, we explore two different initializations: (1) *detection-based initialization:* use multi-view pre-enrollment images to search for the initial bounding boxes where object instances first appear in the video and initialize SOT trackers with the predicted 2D boxes. (2) *template-based initialization:* directly adopt multi-view pre-enrollment images as visual templates in the tracker and set the initial tracking search region to the entire frame.

Implementation details. The cosine similarity threshold in the SAM+DINOv2 approach is 0.6, i.e., the object is considered not visible if the cosine similarity is smaller than the threshold. For a fair comparison, we add additional 2D prediction filtering when re-purposing SOT trackers. We discard 2D predictions from SOT trackers whose prediction scores are lower than 75% of the maximum prediction score. When tracking with MVPE, we first preprocess the captured multi-view images by segmenting and cropping the foreground object using [34]. Many transformer-based SOT trackers only encode a limited number of templates, therefore, we choose 5 images from 0°, 90°, 180°, 270° and top-down for all models in MVPE experiments. We include an ablation study exploring the relationship between tracking results and the number of views used in the supplementary material. To keep the comparison fair, all detection-based trackers in MVPE use SAM+DINOv2 with the same cosine thresholds to locate the initial bounding boxes. In terms of benchmarking RGB-D trackers in MVPE, we utilize the

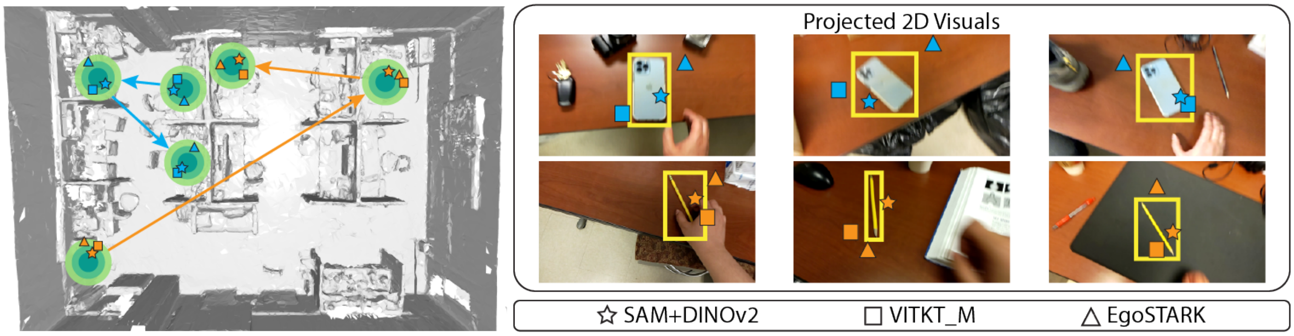

Figure 3. **Qualitative visualizations of tracking with SVOE in both 3D space (left) and projected 2D view (right).** We visualize three top-performing trackers from different categories, i.e., EgoSTARK, VITKT_M, and SAM+DINOv2. For projected 2D visualization, we compare the projected 3D points of each model w.r.t to the ground-truth annotated 2D bounding boxes. In the 3D view, we show 3 concentric circles at each ground-truth position representing 0.25, 0.5 and 0.75 meter thresholds. In both 2D and 3D visualizations, we find SAM+DINOv2 outperforms others as the predictions are closer to the center of object instances.

estimated sparse depth maps using COLMAP [55]. The L2 distance threshold of resetting Kalman filters is 0.15m. All experiments are implemented with PyTorch and run on Nvidia 2080Ti GPUs.

5.1. Benchmark Results

Tracking with SVOE. From the results shown in Table 2, we have the following salient insights: (1) *Re-identifying object instances is important.* Trackers designed with strong re-identify ability, i.e., long-term and egocentric specific types, outperform short-term trackers. Similar findings are shown in recent 2D egocentric tracking work [15, 60]. Surprisingly, SAM+DINOv2, the non-learned approach which does not exploit temporal information beyond the memory heuristic, performs the best among all baselines. We believe the exhaustive proposals on every frame and high quality features provide the model strong, generic re-identification ability. (2) *Depth information is not fully leveraged.* Current RGB-D trackers show similar or slightly worse performance compared to RGB-ST trackers (c.f. MixFormer and MixforRGBD). The main reason is that RGB-D trackers only encode depth maps as auxiliary visual features, which cannot fully exploit the geometric information from depth maps. Additionally, the depth maps are sparse and not always perfectly aligned with RGB images due to camera distortions. (3) *Simple Kalman filter brings marginal benefits.* The Kalman filter does not improve over the simple "most recent" memory heuristic for stationary objects. The naive filter is also not sufficient for modeling the switching between stationary and dynamic motions needed to capture user-object interactions.

Tracking with MVPE. We benchmark top-performing trackers in each category in Table 2 for MVPE setup. From the results shown in Table 3, we find: (1) *SOT trackers cannot fully exploit pre-enrollment information.* SOT methods rely on the initial position defined by 2D boxes on the frame to perform well. Comparing detection-based initializations and SVOE results, e.g., ARTrack^{D} and ARTrack in Table 2, the model performance drops since the initial boxes are not as accurate as ground-truth initialization. VITKT_M adopts many complicated modules that all rely on the initial bounding boxes and degrades more significantly, compared to other types of trackers. (2) *Encoding rich visual information generally helps.* From the results of template-based initializations, VITKT_M for the same reason mentioned before, we find trackers benefit from the high-resolution multi-view images. SAM+DINOv2 shows a significant performance boost because it is more robust to inaccurate initialization without relying on temporal information.

Qualitative results. Figure 3 shows predictions of top-performing trackers from three different categories, i.e., best tracker in long-term and egocentric specific, and SAM+DINOv2. Clearly, SAM+DINOv2 predictions are closer to the object center in both 3D and projected 2D space.

5.2. Further Analysis and Ablation Study

We further compare tracking object instances in both 2D and 3D settings, demonstrating tracking object instances is much easier in 3D space. We also include an ablation study regarding the cosine similarity thresholds. All studies shown in this section are using SAM+DINOv2 unless otherwise specified. More quantitative and qualitative results are shown in the supplementary materials.

Tracking in 2D with 3D guidance. We experimentally demonstrate the importance of leveraging 3D information in egocentric instance tracking by comparing 2D tracking results w/ and w/o 3D guidance. With 3D guidance means the 2D tracking results are computed with *predicted* 3D trajectories as the guidance. For each object instance, the per-frame

Table 2. **Benchmark results of tracking with SVOE.** From the results, we draw three salient conclusions: (1) The ability of re-identifying object instances after they disappear is important, as long-term and egocentric specific trackers outperform short-term trackers, i.e., RGB-ST and RGB-D. (2) Currently, encoding depth maps as auxiliary information cannot improve performance since depth maps are sparse and not always perfectly aligned with RGB frames due to distortions. (3) The Kalman filter smoothing yields marginal improvements over the simple memory heuristic. The method with KF subscript indicates it applies the Kalman filter.

Model	Modality	Precision(%)↑					Recall(%)↑					L2↓	Angle↓
		0.25	0.5	0.75	1.0	1.5	0.25	0.5	0.75	1.0	1.5	(m)	(rad)
ToMP	RGB-ST	5.6	10.1	17.2	25.3	39.0	6.1	11.0	18.8	27.7	42.6	2.11	1.32
MixFormer	RGB-ST	8.3	12.2	18.7	27.0	43.0	9.0	13.4	20.4	29.5	47.0	1.97	1.15
ARTrack	RGB-ST	9.1	13.9	21.5	30.3	45.1	10.1	15.3	23.7	32.4	47.2	1.92	1.10
SAMF	RGB-D	7.0	11.5	15.7	24.0	40.8	7.7	12.5	17.2	26.3	44.7	1.90	1.00
MixForRGBD	RGB-D	7.5	12.1	16.8	25.3	41.0	8.3	13.4	20.1	28.5	45.0	2.11	1.32
ViPT	RGB-D	8.9	13.6	20.6	28.1	41.4	9.7	14.9	22.5	30.7	45.3	2.02	1.21
mixLT	RGB-LT	14.4	17.5	23.9	31.8	47.2	15.8	19.2	26.1	34.8	51.6	1.85	1.02
mlpLT	RGB-LT	16.0	20.0	25.5	35.2	48.2	16.7	20.8	26.5	36.7	50.1	1.77	0.97
VITKT_M	RGB-LT	21.5	24.2	29.7	37.5	50.6	23.0	25.9	31.8	40.2	54.2	1.55	0.83
EgoSTARK	RGB-Ego	17.5	21.2	26.8	36.3	49.1	17.6	22.0	27.4	38.0	51.2	1.70	0.91
SAM + DINOv2	RGB	23.3	26.4	33.1	43.3	59.4	24.9	28.1	35.3	46.3	63.4	1.35	0.81
SAM + DINOv2$_{KF}$	RGB	**23.7**	**27.1**	**33.9**	**44.5**	**61.2**	**25.5**	**29.0**	**36.8**	**48.0**	**64.9**	**1.32**	**0.79**

Table 3. **Benchmark results of tracking with MVPE.** We evaluate top-performing trackers in each category in Table 2 for MVPE setup. From the results, we have the following summaries: (1) *SOT trackers cannot fully exploit pre-enrollment information.* Detection-initialized versions perform less well compared to SVOE due to the inaccurate estimated initial bounding boxes. VITKT_M, which uses many modules that rely heavily on the initialization, degrades more significantly. (2) *Encoding rich visual information generally helps.* SAM+DINOv2 shows an even larger performance boost because it is more robust to the inaccurate initialization. The D and T superscripts indicate the detection- and template-based initializations, respectively.

Model	Modality	Precision(%)↑					Recall(%)↑					L2↓	Angle↓
		0.25	0.5	0.75	1.0	1.5	0.25	0.5	0.75	1.0	1.5	(m)	(rad)
ARTrack^{D}	RGB-ST	6.8	12.1	18.5	25.8	41.0	7.1	12.3	18.8	26.8	42.1	1.98	1.16
ARTrack^{T}	RGB-ST	11.2	18.1	25.2	28.7	38.5	12.7	20.9	28.5	33.0	45.1	1.91	1.07
ViPT^{D}	RGB-D	6.3	11.7	17.9	24.9	40.2	6.9	11.8	17.9	25.9	41.0	2.01	1.21
ViPT^{T}	RGB-D	10.5	17.4	24.0	27.1	36.3	11.9	20.1	27.1	31.3	44.0	1.93	1.10
VITKT_M^{D}	RGB-LT	13.8	18.0	25.0	33.0	46.6	14.3	18.6	25.8	34.1	48.2	1.77	0.98
VITKT_M^{T}	RGB-LT	9.2	12.8	20.7	28.5	44.0	9.7	14.2	22.4	31.3	46.5	1.95	1.08
EgoSTARK^{D}	RGB-Ego	13.2	17.0	23.1	30.5	47.0	14.7	18.5	25.9	34.4	49.7	1.82	1.01
EgoSTARK^{T}	RGB-Ego	18.9	23.1	28.3	37.1	49.6	19.1	23.1	29.3	39.1	52.9	1.67	0.88
SAM + DINOv2	RGB	56.0	59.0	61.8	67.5	74.3	50.0	52.7	55.2	60.3	66.4	0.67	0.40
SAM + DINOv2 $_{KF}$	RGB	**56.2**	**59.4**	**62.2**	**68.1**	**74.8**	**50.3**	**53.1**	**55.7**	**61.1**	**67.1**	**0.65**	**0.39**

Table 4. **Quantitative comparisons of 2D tracking results w/ and w/o 3D guidance.** With 3D guidance means the 2D results are computed by finding the bounding box proposal with the smallest L2 distance from projected 3D trajectories. Without 3D guidance means proposals are selected purely based on the visual feature cosine similarity. Please refer to Section 5.2 for more details. From the results, we find the tracking results are significantly improved with the 3D guidance, indicating that tracking in 3D in egocentric videos is much easier than in 2D by leveraging camera pose and depth sensors.

	3D Guid.	AUC(%)↑	N. Prec.(%)↑	Prec.(%)↑
SVOE	✗	20.7	14.9	8.9
	✓	**27.6**	**21.7**	**11.5**
MVPE	✗	14.1	7.4	3.0
	✓	**39.1**	**35.2**	**18.7**

2D detection results are computed by selecting the (above threshold) proposal with the smallest L2 distance between projected 3D points and the center of bounding boxes proposals. Without 3D guidance means the 2D tracking results are produced by selecting the proposal with the highest cosine feature similarity. To keep a fair comparison, the cosine similarity threshold is the same when computing the 3D and 2D trajectories. We evaluate the 2D tracking performance using widely adopted precision, normalized precision metrics and AUC in SOT literature [69]. As shown in Table 4, the model with 3D guidance performs significantly better in both SVOE and MVPE, demonstrating that leveraging the 3D information, such as camera pose and depth map, makes the tracking problem much easier.

Performance w.r.t cosine similarity thresholds. Feature cosine similarity threshold is adopted to determine whether

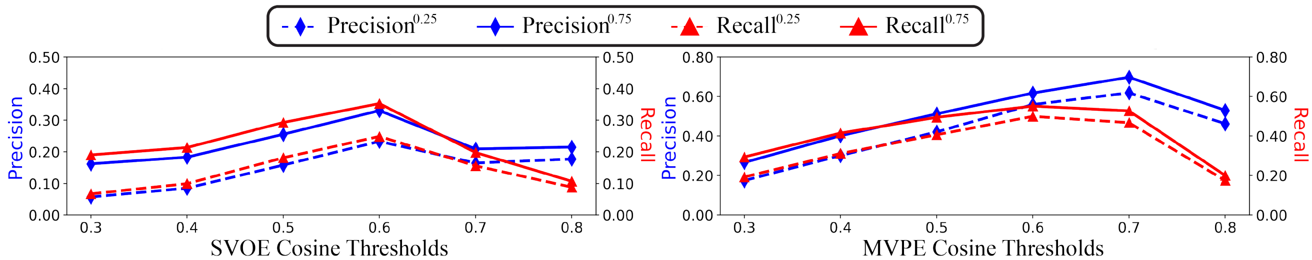

Figure 4. **Performance comparisons of SAM+DINOv2 with different cosine thresholds.** By increasing the threshold, we find the model performance first improves and then gradually decreases. Intuitively, increasing the threshold will initially filter noisy predictions but when the threshold is too large the model will miss correct object 3D location updates.

the object instance is present in the current frame, which is crucial for the memory updating mechanism. To characterize the relationship between tracking performance and cosine similarity thresholds, we run the experiment with different cosine similarity thresholds but keep everything else the same. As shown in Figure 4, both models show improved performance at first and then a gradual decrease. Higher cosine thresholds result in fewer predictions so the model must increasingly rely on previous confident predictions stored in the memory. Models with large cosine similarity thresholds have a higher chance of missing valid location updates, which leads to a drop in both precision and recall.

6. Discussion

Limitations and future work. We point out that the current benchmark dataset has limited geographic and demographic diversity and captures only a small range of objects and activities. As such it is not appropriate for training large models and only serves as a diagnostic test to identify some limitations of existing approaches. Our hope is that it serves as a starting point for the research community to explore and eventually grow into a more comprehensive challenge. Currently, the studied baseline approaches follow the same paradigm, i.e., lifting predicted 2D trajectories into 3D space. We found empirically that the simplest memory mechanism performed best but it seems very likely there are more nuanced state-update models which can integrate multiple observations effectively.

Finally, we highlight two opportunities for future work. First, advanced models to detect object 3D motion changes. Our experiments demonstrate that tracking in 3D world coordinates effectively narrows the problem to that of accurately predicting the object motion status, i.e., finding all stationary periods for each object instance. However, accurately predicting object state changes is still a non-trivial problem to solve. Second, better utilization of object instance information. Currently, the object instances enrollments, i.e., SVOE and MVPE are naively encoded as visual features. Future work should explore the approaches of fusing the additional scene 3D information with object instances for better tracking performance.

Broader impact. We believe the broader impact of our work is two-fold. First, we hope our benchmark brings more attention to the problem of tracking object instances in 3D from the egocentric perspective and contributes towards building future task-aware assistive agents. Second, our multi-modal benchmark dataset is beneficial to the study of other 3D scene understanding related problems from the egocentric perspective, such as SLAM, camera localization, 3D reconstruction, and depth estimation.

Potential negative impacts. Tracking in 3D from egocentric videos requires the geometric data of surrounding environments and the sensor streams that continuously capture their workplace or daily lives. There are obvious privacy concerns when deploying such hardware and algorithms. Similar to other apps running on personal devices, the simple solution is to keep all user data locally or (in the context of research) develop techniques for anonymizing video [61].

7. Conclusion

We introduce a new *IT3DEgo* benchmark that allows us to study the problem of tracking object instances in 3D from egocentric videos. The object instances to be tracked are either determined in advance or enrolled online during user interactions with the environment. To support the study, we collect and annotate a new dataset that features RGB-D videos and per-frame camera poses, along with instance-level annotations in both 2D camera and 3D world coordinate frames. We re-purpose and evaluate state-of-the-art single object trackers and develop a strong baseline using large pretrained recognition models and Kalman filtering. We hope our benchmark brings more attention to this challenge and contributes to the development of perceptually-aware assistive agents.

Acknowledgements. This work was partially supported by the DARPA Perceptually enabled Task Guidance (PTG) Program (HR00112220005). Shu Kong was partially supported by the University of Macau (SRG2023-00044-FST). We thank Juan Carlos Dibene for the help of using hl2ss [1]; An Dang for the help with the initial development of the data capture; Yi-Ling Lin for the help of partial data annotations.

References

[1] Hololens 2 sensor streaming. https://github.com/jdibenes/hl2ss, 2023. 8

[2] Adel Ahmadyan, Liangkai Zhang, Artsiom Ablavatski, Jianing Wei, and Matthias Grundmann. Objectron: A large scale dataset of object-centric videos in the wild with pose annotations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7822–7831, 2021. 2

[3] Goutam Bhat, Martin Danelljan, Luc Van Gool, and Radu Timofte. Learning discriminative model prediction for tracking. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6182–6191, 2019. 4

[4] Huikun Bi, Ruisi Zhang, Tianlu Mao, Zhigang Deng, and Zhaoqi Wang. How can i see my future? fvtraj: Using first-person view for pedestrian trajectory prediction. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VII 16*, pages 576–593. Springer, 2020. 1

[5] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11621–11631, 2020. 2

[6] Yukang Chen, Jianhui Liu, Xiangyu Zhang, Xiaojuan Qi, and Jiaya Jia. Voxelnext: Fully sparse voxelnet for 3d object detection and tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21674–21683, 2023. 2

[7] Yutao Cui, Cheng Jiang, Limin Wang, and Gangshan Wu. Mixformer: End-to-end tracking with iterative mixed attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13608–13618, 2022. 5

[8] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5828–5839, 2017. 2

[9] Dima Damen, Teesid Leelasawassuk, Osian Haines, Andrew Calway, and Walterio W Mayol-Cuevas. You-do, i-learn: Discovering task relevant objects and their modes of interaction from multi-user egocentric video. In *BMVC*, page 3, 2014. 2

[10] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, et al. Scaling egocentric vision: The epic-kitchens dataset. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 720–736, 2018. 2, 3

[11] Martin Danelljan, Luc Van Gool, and Radu Timofte. Probabilistic regression for visual tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7183–7192, 2020. 4

[12] Samyak Datta, Sameer Dharur, Vincent Cartillier, Ruta Desai, Mukul Khanna, Dhruv Batra, and Devi Parikh. Episodic memory question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19119–19128, 2022. 3

[13] Mariella Dimiccoli, Juan Marín, and Edison Thomaz. Mitigating bystander privacy concerns in egocentric activity recognition with deep learning and intentional image degradation. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 1(4):1–18, 2018. 2

[14] Matteo Dunnhofer, Antonino Furnari, Giovanni Maria Farinella, and Christian Micheloni. Is first person vision challenging for object tracking? In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2698–2710, 2021. 3

[15] Matteo Dunnhofer, Antonino Furnari, Giovanni Maria Farinella, and Christian Micheloni. Visual object tracking in first person vision. *International Journal of Computer Vision*, 131(1):259–283, 2023. 1, 2, 6

[16] Debidatta Dwibedi, Ishan Misra, and Martial Hebert. Cut, paste and learn: Surprisingly easy synthesis for instance detection. In *Proceedings of the IEEE international conference on computer vision*, pages 1301–1310, 2017. 3

[17] Alircza Fathi, Jessica K Hodgins, and James M Rehg. Social interactions: A first-person perspective. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1226–1233. IEEE, 2012. 2

[18] Basura Fernando and Samitha Herath. Anticipating human actions by correlating past with the future with jaccard similarity measures. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13224–13233, 2021. 2

[19] Antonino Furnari and Giovanni Maria Farinella. What would you expect? anticipating egocentric actions with rolling-unrolling lstms and modality attention. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6252–6261, 2019. 2

[20] Mathieu Garon and Jean-François Lalonde. Deep 6-dof tracking. *IEEE transactions on visualization and computer graphics*, 23(11):2410–2418, 2017. 2

[21] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research*, 32(11):1231–1237, 2013. 2

[22] Georgios Georgakis, Md Alimoor Reza, Arsalan Mousavian, Phi-Hung Le, and Jana Košecká. Multiview rgb-d dataset for object instance detection. In *2016 Fourth International Conference on 3D Vision (3DV)*, pages 426–434. IEEE, 2016. 3

[23] Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, et al. Ego4d: Around the world in 3,000 hours of egocentric video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18995–19012, 2022. 1, 2, 3

[24] Tomáš Hodaň, Vibhav Vineet, Ran Gal, Emanuel Shalev, Jon Hanzelka, Treb Connell, Pedro Urbina, Sudipta N Sinha, and Brian Guenter. Photorealistic image synthesis for object instance detection. In *2019 IEEE international conference on image processing (ICIP)*, pages 66–70. IEEE, 2019. 3

[25] Hao Jiang and Kristen Grauman. Seeing invisible poses: Estimating 3d body pose from egocentric video. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3501–3509. IEEE, 2017. 2

[26] Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. 1960. 2, 5

[27] Evangelos Kazakos, Arsha Nagrani, Andrew Zisserman, and Dima Damen. Epic-fusion: Audio-visual temporal binding for egocentric action recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5492–5501, 2019. 2

[28] Aleksandr Kim, Aljoša Ošep, and Laura Leal-Taixé. Eagermot: 3d multi-object tracking via sensor fusion. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 11315–11321. IEEE, 2021. 2

[29] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. *arXiv preprint arXiv:2304.02643*, 2023. 2, 3, 5

[30] Matej Kristan, JirıMatas, Aleš Leonardis, Michael Felsberg, Roman Pflugfelder, Joni-Kristian Kamarainen, Hyung Jin Chang, Martin Danelljan, Luka Čehovin Zajc, Alan Lukežič, Ondrej Drbohlav, Jani Kapyla, Gustav Hager, Song Yan, Jinyu Yang, Zhongqun Zhang, Gustavo Fernandez, and et. al. The ninth visual object tracking vot2021 challenge results, 2021. 5

[31] Matej Kristan, Ales Leonardis, Jiri Matas, Michael Felsberg, Roman Pflugfelder, Joni-Kristian Kamarainen, Hyung Jin Chang, Martin Danelljan, Luka Čehovin Zajc, Alan Lukežič, Ondrej Drbohlav, Johanna Bjorklund, Yushan Zhang, Zhongqun Zhang, Song Yan, Wenyan Yang, Dingding Cai, Christoph Mayer, and Gustavo Fernandez. The tenth visual object tracking vot2022 challenge results, 2022. 5

[32] Yong Jae Lee, Joydeep Ghosh, and Kristen Grauman. Discovering important people and objects for egocentric video summarization. In *2012 IEEE conference on computer vision and pattern recognition*, pages 1346–1353. IEEE, 2012. 2

[33] Haoxin Li, Yijun Cai, and Wei-Shi Zheng. Deep dual relation modeling for egocentric interaction recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7932–7941, 2019. 2

[34] Jiachen Li, Jitesh Jain, and Humphrey Shi. Matting anything. *arXiv: 2306.05399*, 2023. 5

[35] Yiming Li, Ziang Cao, Andrew Liang, Benjamin Liang, Luoyao Chen, Hang Zhao, and Chen Feng. Egocentric prediction of action target in 3d. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 20971–20980. IEEE, 2022. 3

[36] Miao Liu, Siyu Tang, Yin Li, and James M Rehg. Forecasting human-object interaction: joint prediction of motor attention and actions in first person video. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16*, pages 704–721. Springer, 2020. 2

[37] Yunze Liu, Yun Liu, Che Jiang, Kangbo Lyu, Weikang Wan, Hao Shen, Boqiang Liang, Zhoujie Fu, He Wang, and Li Yi. Hoi4d: A 4d egocentric dataset for category-level human-object interaction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21013–21022, 2022. 2

[38] Zhijian Liu, Haotian Tang, Alexander Amini, Xinyu Yang, Huizi Mao, Daniela L Rus, and Song Han. Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2774–2781. IEEE, 2023. 2

[39] Christoph Mayer, Martin Danelljan, Goutam Bhat, Matthieu Paul, Danda Pani Paudel, Fisher Yu, and Luc Van Gool. Transforming model prediction for tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8731–8740, 2022. 5

[40] Evonne Ng, Donglai Xiang, Hanbyul Joo, and Kristen Grauman. You2me: Inferring body pose in egocentric video via first and second person interactions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9890–9900, 2020. 2

[41] Khanh Nguyen and Hal Daumé III. Help, anna! visual navigation with natural multimodal assistance via retrospective curiosity-encouraging imitation learning. *arXiv preprint arXiv:1909.01871*, 2019. 1

[42] Curtis Northcutt, Shengxin Zha, Steven Lovegrove, and Richard Newcombe. Egocom: A multi-person multi-modal egocentric communications dataset. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020. 2

[43] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 2, 5

[44] Xiaqing Pan, Nicholas Charron, Yongqian Yang, Scott Peters, Thomas Whelan, Chen Kong, Omkar Parkhi, Richard Newcombe, and Yuheng Carl Ren. Aria digital twin: A new benchmark dataset for egocentric 3d machine perception. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 20133–20143, 2023. 3

[45] Hitesh A Patel and Darshak G Thakore. Moving object tracking using kalman filter. *International Journal of Computer Science and Mobile Computing*, 2(4):326–332, 2013. 5

[46] Hamed Pirsiavash and Deva Ramanan. Detecting activities of daily living in first-person camera views. In *2012 IEEE conference on computer vision and pattern recognition*, pages 2847–2854. IEEE, 2012. 2

[47] Chiara Plizzari, Mirco Planamente, Gabriele Goletto, Marco Cannici, Emanuele Gusso, Matteo Matteucci, and Barbara Caputo. E2 (go) motion: Motion augmented event stream for egocentric action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19935–19947, 2022. 2

[48] Rafael Possas, Sheila Pinto Caceres, and Fabio Ramos. Egocentric activity recognition on a budget. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5967–5976, 2018. 2

[49] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning

transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 3

[50] Vijay Raghavan, Peter Bollmann, and Gwang S Jung. A critical investigation of recall and precision as measures of retrieval system performance. *ACM Transactions on Information Systems (TOIS)*, 7(3):205–229, 1989. 3

[51] Nicholas Rhinehart and Kris M Kitani. First-person activity forecasting with online inverse reinforcement learning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3696–3705, 2017. 2

[52] Grégory Rogez, James S Supancic, and Deva Ramanan. First-person pose recognition using egocentric workspaces. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4325–4333, 2015. 2

[53] Michael Ryoo, Brandon Rothrock, Charles Fleming, and Hyun Jong Yang. Privacy-preserving human activity recognition from extreme low resolution. In *Proceedings of the AAAI conference on artificial intelligence*, 2017. 2

[54] Michael S Ryoo and Larry Matthies. First-person activity recognition: What are they doing to me? In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2730–2737, 2013. 2

[55] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 6

[56] Qianqian Shen, Yunhan Zhao, Nahyun Kwon, Jeeeun Kim, Yanan Li, and Shu Kong. A high-resolution dataset for instance detection with multi-view object capture. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023. 3

[57] Hangjie Shi, Leslie Ball, Govind Thattai, Desheng Zhang, Lucy Hu, Qiaozi Gao, Suhaila Shakiah, Xiaofeng Gao, Aishwarya Padmakumar, Bofei Yang, et al. Alexa, play with robot: Introducing the first alexa prize simbot challenge on embodied ai. *arXiv preprint arXiv:2308.05221*, 2023. 1

[58] Yu-Chuan Su and Kristen Grauman. Detecting engagement in egocentric video. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part V 14*, pages 454–471. Springer, 2016. 2

[59] Andrew Szot, Alexander Clegg, Eric Undersander, Erik Wijmans, Yili Zhao, John Turner, Noah Maestre, Mustafa Mukadam, Devendra Singh Chaplot, Oleksandr Maksymets, et al. Habitat 2.0: Training home assistants to rearrange their habitat. *Advances in Neural Information Processing Systems*, 34:251–266, 2021. 1

[60] Hao Tang, Kevin Liang, Kristen Grauman, Matt Feiszli, and Weiyao Wang. Egotracks: A long-term egocentric visual object tracking dataset. *arXiv preprint arXiv:2301.03213*, 2023. 1, 2, 3, 5, 6

[61] Daksh Thapar, Aditya Nigam, and Chetan Arora. Anonymizing egocentric videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2320–2329, 2021. 2, 8

[62] Dorin Ungureanu, Federica Bogo, Silvano Galliani, Pooja Sama, Xin Duan, Casey Meekhof, Jan Stühmer, Thomas J Cashman, Bugra Tekin, Johannes L Schönberger, et al. Hololens 2 research mode as a tool for computer vision research. *arXiv preprint arXiv:2008.11239*, 2020. 3

[63] Guangting Wang, Chong Luo, Xiaoyan Sun, Zhiwei Xiong, and Wenjun Zeng. Tracking by instance detection: A meta-learning approach. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6288–6297, 2020. 3

[64] Jian Wang, Lingjie Liu, Weipeng Xu, Kripasindhu Sarkar, Diogo Luvizon, and Christian Theobalt. Estimating egocentric 3d human pose in the wild with external weak supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13157–13166, 2022. 1, 2

[65] Xing Wei, Yifan Bai, Yongchao Zheng, Dahu Shi, and Yihong Gong. Autoregressive visual tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9697–9706, 2023. 5

[66] Shiuh-Ku Weng, Chung-Ming Kuo, and Shu-Kang Tu. Video object tracking using adaptive kalman filter. *Journal of Visual Communication and Image Representation*, 17(6):1190–1208, 2006. 5

[67] Xinshuo Weng, Jianren Wang, David Held, and Kris Kitani. 3d multi-object tracking: A baseline and new evaluation metrics. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 10359–10366. IEEE, 2020. 2, 5

[68] Jimmy Wu, Rika Antonova, Adam Kan, Marion Lepert, Andy Zeng, Shuran Song, Jeannette Bohg, Szymon Rusinkiewicz, and Thomas Funkhouser. Tidybot: Personalized robot assistance with large language models. *arXiv preprint arXiv:2305.05658*, 2023. 1

[69] Yi Wu, Jongwoo Lim, and Ming-Hsuan Yang. Online object tracking: A benchmark. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2411–2418, 2013. 7

[70] Bin Yan, Houwen Peng, Jianlong Fu, Dong Wang, and Huchuan Lu. Learning spatio-temporal transformer for visual tracking. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10448–10457, 2021. 4

[71] Mang Ye, Jianbing Shen, Gaojie Lin, Tao Xiang, Ling Shao, and Steven CH Hoi. Deep learning for person re-identification: A survey and outlook. *IEEE transactions on pattern analysis and machine intelligence*, 44(6):2872–2893, 2021. 3

[72] Alper Yilmaz, Omar Javed, and Mubarak Shah. Object tracking: A survey. *Acm computing surveys (CSUR)*, 38(4):13–es, 2006. 3

[73] Tianwei Yin, Xingyi Zhou, and Philipp Krahenbuhl. Center-based 3d object detection and tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11784–11793, 2021. 2

[74] Ryo Yonetani, Kris M Kitani, and Yoichi Sato. Recognizing micro-actions and reactions from paired egocentric videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2629–2638, 2016. 2

[75] Yunhan Zhao, Shu Kong, Daeyun Shin, and Charless Fowlkes. Domain decluttering: Simplifying images to mitigate synthetic-real domain shift and improve depth estimation.

In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3330–3340, 2020. 3

[76] Yunhan Zhao, Shu Kong, and Charless Fowlkes. Camera pose matters: Improving depth prediction by mitigating pose distribution bias. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15759–15768, 2021. 3

[77] Liang Zheng, Yi Yang, and Alexander G Hauptmann. Person re-identification: Past, present and future. *arXiv preprint arXiv:1610.02984*, 2016. 3

[78] Changqing Zhou, Zhipeng Luo, Yueru Luo, Tianrui Liu, Liang Pan, Zhongang Cai, Haiyu Zhao, and Shijian Lu. Pttr: Relational 3d point cloud object tracking with transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8531–8540, 2022. 2

[79] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. *arXiv preprint arXiv:1904.07850*, 2019. 2

[80] Jiawen Zhu, Simiao Lai, Xin Chen, Dong Wang, and Huchuan Lu. Visual prompt multi-modal tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9516–9526, 2023. 5