Instance Tracking in 3D Scenes from Egocentric Videos
–Supplementary Material–

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

Outline

This document supplements the main paper with additional details of the benchmark dataset, more experimental results, dataset documents, and visualizations. Below is the outline of this document.

- **Section 1**. Additional details of the benchmark dataset, including the capture procedure of raw videos and object instances, and the annotation steps.
- **Section 2**. We conduct extensive studies and analyses on the improved baseline approach, SAM+DINOv2.
- **Section 3**. Dataset documentation and intended uses.
- **Section 4**. Visualizations of 2D frames from raw video sequences and 3D meshes of the capture environments.

1. Additional Dataset Details

We present additional details of the datasets, such as collection details and annotations, to help others better understand and utilize the benchmark dataset. Note that the data collection protocol was registered with the appropriate institutional review board (IRB).

Raw video collections. We capture the raw data using HoloLens2 that includes 1 RGB camera, 4 grayscale cameras, and 1 depth sensor operating in 2 different modes, shown in Figure 1. Considering the downstream application scenarios of our benchmark task, we choose to capture our benchmark dataset in 10 different indoor scenes. To capture the real-time geometry information, we capture all videos with high fps AHAT depth mode in HoloLens2 [7]. Note that AHAT depth maps come with phase wrapping [6] at 1 meter but they can be unwrapped using rendered depth from mesh or exploring existing unwrapping algorithms [3, 4]. Before capturing in a new environment, we have a warm-up phase to make the device familiar with the surrounding environment in order to output accurate camera poses when capturing the video. In the warm-up phase, we walk around in the environment with the HoloLens2 turned on and make sure the device has seen all visible surfaces. In practice, we spend around 20 minutes for the warm-up phase when we move to a new environment and around 5 minutes every time before we capture the new video.

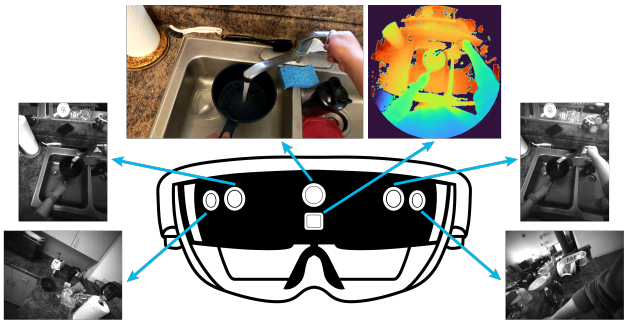

Figure 1. **Illustration of our benchmark dataset.** It is collected with HoloLens2 which captures RGB, depth, and four grayscale side views at 30 fps. Additionally, the device also captures per-frame camera poses allowing coarse reconstruction of the surroundings.

Object instance collections. The entire videos come with 220 unique object instances, which cover a wide range of object instances for naturalistic daily tasks, such as cooking, writing, and repairing. For each instance, we take 25 high-resolution images on a rotary table with the QR code (c.f. Figure 2 for visual examples). Specifically, the photos are taken by hand-held iPhone 13 Pro approximately 45 cm away from the object center. As illustrated in Figure 3, 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 zoom in 2.5 times for objects whose diameter is lower than 20 cm to ensure the object instance is large enough in the image and use the normal scale (no zoom) for the rest of the case.

Annotations. There are three types of manual annotations along with our benchmark dataset. First, the 3D center of each object instance. The annotator is first asked to draw boxes on depth maps from $\geq$5 diverse views if possible. Each 2D bounding box is lifted to the 3D space with camera poses. The 3D centers of each object instance in a stationary period are averaged to get the initial estimation. The annota-

Figure 2. **Visualization of raw and preprocessed multi-view images**. Raw images represent the images directly output from the capture device, i.e., iPhone 13 Pro. We process raw images with segmentation and cropping before feeding them into the models. For more implementation details, please check Section 5 in the main paper.

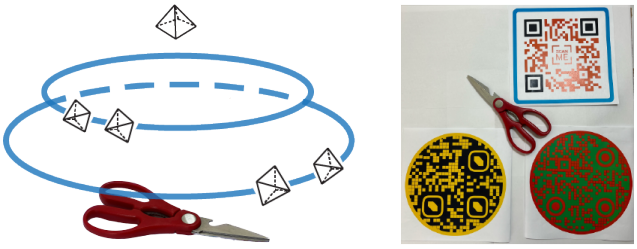

Figure 3. **Illustration of our multi-view capture setup.** The left panel shows our camera positions when taking 25 images to support the pre-enrollment study. Specifically, we take 12 object-centric photos evenly from 360° while keeping the camera 30° elevation. Another 12 images are taken in a similar fashion while keeping the camera 60° elevation. Lastly, we take one top-down view. An example of the top-down view with the QR code is shown on the right.

tors then examine the adjust the annotated 3D points based on the RGB frames from the video sequence and captured mesh. Second, the 2D axis-align bounding boxes of each object instance every five frames starting from the beginning of the video. Specifically, we ask the annotators to go through the entire video first. We provide one video frame with a 2D bounding box to specify each object instance to the annotators. We ask annotators to draw *amodal* bounding boxes of each object instance and do not annotate the object instances with heavy occlusions (i.e., when less than 25% of the object is visible). The last type of annotation is the object motion state. The object is annotated as stationary only when the hands are no longer in contact with the object. All annotations are first labeled by a group of annotators and checked by other independent annotators to ensure the quality.

2. Additional Ablation study

This section supplements the results in the main paper with the following 4 experiments. We analyze the performance change w.r.t the number of views used in MVPE, memory update mechanism, feature encoder, and proposal generator of the improved baseline, i.e., SAM+DINOv2. All experiments use online enrollment (SVOE) except the number of views study.

Performance w.r.t number of views. Due to the architecture design of many transformer-based trackers, we only use 5 views in the benchmark experiment. In this section, we further study the relationship between the number of views and the tracking performance. Specifically, we compare

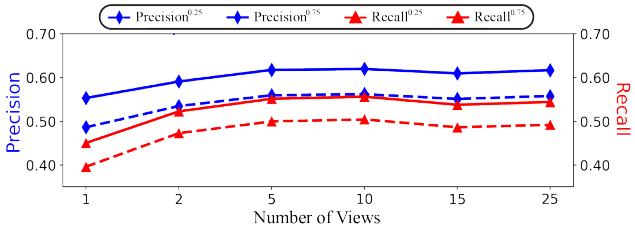

Figure 4. **Performance w.r.t number views in MVPE.** We run SAM+DINOv2 with different numbers of views while keeping everything else the same for a fair comparison. We find the performance saturates after using 5 views. This suggests that simply encode and average features benefit from a higher number of views (i.e., number of views from 1 to 5) but still cannot fully exploit the visual information from different views (i.e., after using 5 views).

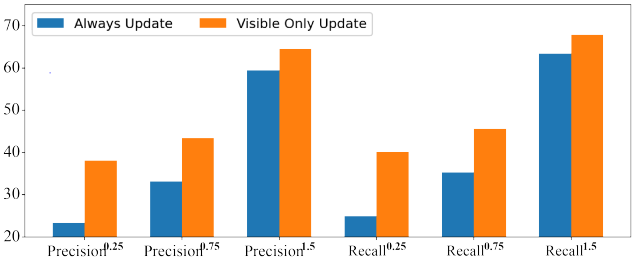

Figure 5. **Performance improvement by updating on visible only frames.** We control the memory update of SAM+DINOv2 by updating the memory only when the object instance is visible. We find the performance is significantly improved, indicating one of the major challenges of the baseline is to correctly update the memory with high quality predictions.

the performance of SAM+DINOv2 with 1, 2, 5, 10, 15, 25 images while keeping all other parameters the same. As shown in Figure 4, the performance improves from 1 view to 5 views but quickly saturates after using 5 views. This suggests that naively encode and average features benefit from a higher number of views but still cannot fully exploit the visual information from different views.

Performance improvement with visible update only. From the results shown in Table 2 and Table 3 in the main paper, we find identifying high quality predictions and updating the memory is the main challenge in the proposed baseline pipeline. To further validate this idea, we control the update of memory in SAM+DINOv2 model by only updating on the visible frames. We extract the visible information from the 2D annotations. In other words, the memory for each instance is only updated on the frame where the 2D bounding box is annotated. As shown in Figure 5, updating the memory only when object instances are visible significantly improves the performance. Although the update timing is correct, errors from 2D predictions, depth maps and camera poses prevent the model from improving further.

Performance w.r.t different feature encoders. The top-performing baseline, i.e., SAM+DINOv2 adopts DINOv2 as the pretrained feature encoder. To further explore the performance w.r.t different large-scale feature encoders, we experiment with another state-of-the-art feature encoder, i.e., DINO [2]. We plot the results using DINO and DINOv2 at different cosine thresholds in Figure 6. From the results, we find: (1) *Stronger encoder improves the performance.* The best performance of SAM+DINOv2 is stronger than SAM+DINO where both models have the peak performance when the cosine threshold equals 0.6. (2) *Similar performance trend w.r.t cosine similarity changes.* The performance of both models first improves and then gradually decreases when increasing the cosine threshold from 0.3 to 0.8.

Table 1. **Quantitative comparisons of different proposal generators.** We compare the performance of SAM+DINOv2 and YOLOv7+DINOv2. To keep the comparison fair, the only differences between these models are the proposal generators. From the results, we find adopting YOLOv7 makes the performance slightly worse. The proposal quality from YOLOv7 is lower but runs faster.

Proposals	Precision(%)↑			Recall(%)↑			L2↓
	0.25	0.75	1.5	0.25	0.75	1.5	(m)
YOLOv7	20.3	28.1	50.2	21.5	30.7	53.9	1.72
SAM	**23.3**	**33.1**	**59.4**	**24.9**	**35.3**	**63.4**	**1.35**

Comparisons of different proposal generators. Currently, the improved baseline utilizes SAM as the proposal generator. In this part, we replace SAM with the proposals from YOLOv7, i.e., the output before the final classification layer. The results are shown in Table 1. Although the performance of YOLOv7+DINOv2 is lower compared to SAM+DINOv2, which is not surprising. The proposal quality from YOLOv7 is lower but runs faster. However, the current baseline approaches are not able to run in real time due to the following encoding and lifting steps. One promising direction for future work is to improve the speed of the tracking models.

3. Datasheet

We follow the datasheet proposed in [5] for documenting our benchmark dataset.

Motivation

For what purpose was the dataset created?
This dataset was created to study the problem of instance tracking in 3D from egocentric videos. We find current egocentric sensor data from AR/VR devices cannot support the study of our benchmark problem.

Composition

What do the instances that comprise the dataset represent?

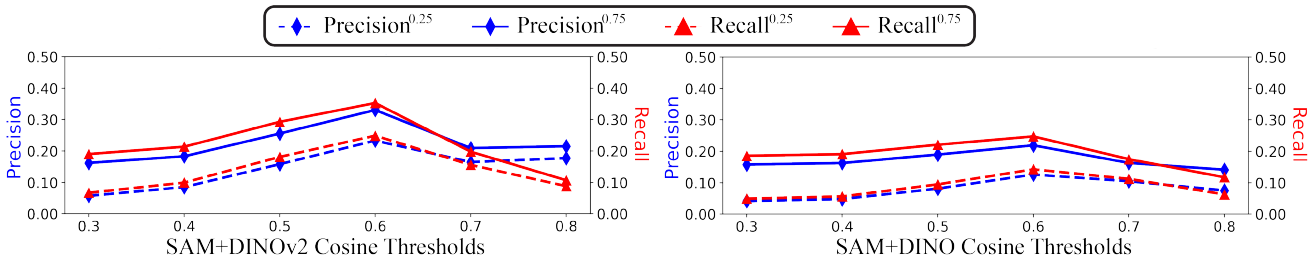

Figure 6. **Performance comparisons of different encoders at various cosine thresholds**. From the results, we find: (1) *Stronger encoder improve the performance.* The best performance of SAM+DINOv2 is stronger than SAM+DINO where both models have the peak performance when the cosine threshold equals 0.6. (2) *Similar performance trend w.r.t cosine similarity changes.* The performance of both models first improves and then gradually decreases when increasing the cosine threshold from 0.3 to 0.8.

Raw egocentric video sequences, object enrollments for each object instance, and annotation files.

How many instances are there in total?

There are 50 video sequences with an average length of over 10K frames, 220 unique object instances with two types of enrollment information, and three types of annotations.

Does the dataset contain all possible instances or is it a sample (not necessarily random) of instances from a larger set?

Yes.

What data does each instance consist of?

Please check Section 3.2 in the main paper for details.

Is there a label or target associated with each instance?

Yes. Please check Section 3.2 in the main paper for details.

Is any information missing from individual instances?

No.

Are relationships between individual instances made explicit?

Videos captured in the same scene share a similar surrounding environment but different activities. Object instances are related to the task performed in the video. No explicit relationships between different object instances in the same video.

Are there recommended data splits?

Yes. The entire benchmark dataset focuses on evaluation only. Models should be pretrained on other data sources. Please check Section 3.1 in the main paper for details.

Are there any errors, sources of noise, or redundancies in the dataset?

Yes. There are noises in camera poses and depth maps. The source of camera pose noise is from the camera localization from HoloLens2, especially under large head motion. The depth map noises are from phase wrapping. But this noise can be easily recovered with rendered depth using mesh or exploring existing unwrapping algorithms.

Is the dataset self-contained, or does it link to or otherwise rely on external resources (e.g., websites, tweets, other datasets)?

Yes. The dataset is self-contained.

Does the dataset contain data that might be considered confidential (e.g., data that is protected by legal privilege or by doctor-patient confidentiality, data that includes the content of individuals' non-public communications)?

No.

Does the dataset contain data that, if viewed directly, might be offensive, insulting, threatening, or might otherwise cause anxiety?

No.

Does the dataset identify any subpopulations (e.g., by age, gender)?

No.

Is it possible to identify individuals (i.e., one or more natural persons), either directly or indirectly (i.e., in combination with other data) from the dataset?

No. We have carefully examined the data and ensure no personally identifiable information is included.

Does the dataset contain data that might be considered sensitive in any way (e.g., data that reveals racial or ethnic origins, sexual orientations, religious beliefs, political opinions or union memberships, or locations; financial or health data; biometric or genetic data; forms of government identification, such as social security numbers; criminal history)?

No..

Any other comments?

N/A

Collection Process

How was the data associated with each instance acquired?

The raw video sequences are collected with HoloLens2. The pre-enrollment information is captured with the iPhone 13 Pro. The rest data, i.e, annotations and online enrollment information, are acquired from human annotators.

What mechanisms or procedures were used to collect the data (e.g., hardware apparatus or sensor, manual human curation, software program, software API)?

The dataset is collected with open-source hl2ss [1] using HoloLens2. The pre-enrollment images are captured with the iPhone 13 Pro. For more details please check Section 3.1

in the paper.

If the dataset is a sample from a larger set, what was the sampling strategy (e.g., deterministic, probabilistic with specific sampling probabilities)?

N/A

Does the dataset relate to people?

Yes. The dataset includes video sequences of the first-person view of individuals performing the daily activity.

Were any ethical review processes conducted (e.g., by an institutional review board)?

Yes. Data collection protocol was registered with the appropriate institutional review board (IRB).

Did you collect the data from the individuals in question directly, or obtain it via third parties or other sources (e.g., websites)?

The raw video sequences are collected when the camera wearer performs the daily task.

Were the individuals in question notified about the data collection?

Yes.

Did the individuals in question consent to the collection and use of their data?

Yes.

If consent was obtained, were the consenting individuals provided with a mechanism to revoke their consent in the future or for certain uses?

No.

Has an analysis of the potential impact of the dataset and its use on data subjects (e.g., a data protection impact analysis) been conducted?

No. All annotations are on objective world states with no subjective opinions or arguments involved.

Any other comments?

N/A

Preprocessing/Cleaning/Labeling

Was any preprocessing/cleaning/labeling of the data done (e.g., discretization or bucketing, tokenization, part-of-speech tagging, SIFT feature extraction, removal of instances, processing of missing values)?

No.

Was the "raw" data saved in addition to the preprocessed/cleaned/labeled data (e.g., to support unanticipated future uses)?

Yes. We will provide both the raw data and annotations.

Is the software used to preprocess/clean/label the instances available?

No.

Any other comments?

N/A

Uses

Has the dataset been used for any tasks already?

No.

What (other) tasks could the dataset be used for?

Our benchmark dataset also supports the study of other 3D scene understanding problems from egocentric videos, such as SLAM, depth estimation, and camera localization.

Is there anything about the composition of the dataset or the way it was collected and preprocessed/cleaned/labeled that might impact future uses?

No.

Are there tasks for which the dataset should not be used?

The usage of this dataset should be limited to the scope of instance tracking in 3D and geometric scene understanding from egocentric videos.

Any other comments?

N/A

Distribution

Will the dataset be distributed to third parties outside of the entity (e.g., company, institution, organization) on behalf of which the dataset was created?

Yes. The dataset will be made publicly available and third parties are allowed to distribute the dataset.

How will the dataset will be distributed (e.g., tarball on website, API, GitHub)?

The dataset will be publicly available on both Github repo and the website and stored on the cloud store, e.g., Google drive or Amazon S3.

When will the dataset be distributed?

The full dataset will be released to the public upon acceptance of this paper.

Will the dataset be distributed under a copyright or other intellectual property (IP) license, and/or under applicable terms of use (ToU)?

We release our benchmark dataset and code under MIT license.

Have any third parties imposed IP-based or other restrictions on the data associated with the instances?

No.

Do any export controls or other regulatory restrictions apply to the dataset or to individual instances?

No.

Any other comments?

N/A

Maintenance

Is there an erratum?

No. When errors are confirmed, we will announce erratum on the platform where dataset is publicly hosted, i.e., either the Github repo or the website.

Will the dataset be updated (e.g., to correct labeling errors, add new instances, delete instances')?

Yes. We hope to bring more diversity to the dataset, such as more object instance and scenes.

If the dataset relates to people, are there applicable limits on the retention of the data associated with the instances (e.g., were individuals in question told that their data would be retained for a fixed period of time and then deleted)?

No.

Will older versions of the dataset continue to be supported/hosted/maintained?

Yes. All versions of the dataset will be publicly available.

If others want to extend/augment/build on/contribute to the dataset, is there a mechanism for them to do so?

Please email us if you are interested in extending or contributing to the dataset.

Any other comments?

N/A

4. Additional Visualizations

We include additional 2D and 3D visualizations of our benchmark dataset in Figure 7.

References

[1] Hololens 2 sensor streaming. https://github.com/jdibenes/hl2ss, 2023. 4

[2] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021. 3

[3] David Droeschel, Dirk Holz, and Sven Behnke. Multi-frequency phase unwrapping for time-of-flight cameras. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1463–1469. IEEE, 2010. 1

[4] David Droeschel, Dirk Holz, and Sven Behnke. Probabilistic phase unwrapping for time-of-flight cameras. In *ISR 2010 (41st International Symposium on Robotics) and ROBOTIK 2010 (6th German Conference on Robotics)*, pages 1–7. VDE, 2010. 1

[5] Timnit Gebru, Jamie Morgenstern, Briana Vecchione, Jennifer Wortman Vaughan, Hanna Wallach, Hal Daumé Iii, and Kate Crawford. Datasheets for datasets. *Communications of the ACM*, 64(12):86–92, 2021. 3

[6] Miles Hansard, Seungkyu Lee, Ouk Choi, and Radu Patrice Horaud. *Time-of-flight cameras: principles, methods and applications*. Springer Science & Business Media, 2012. 1

[7] 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. 1

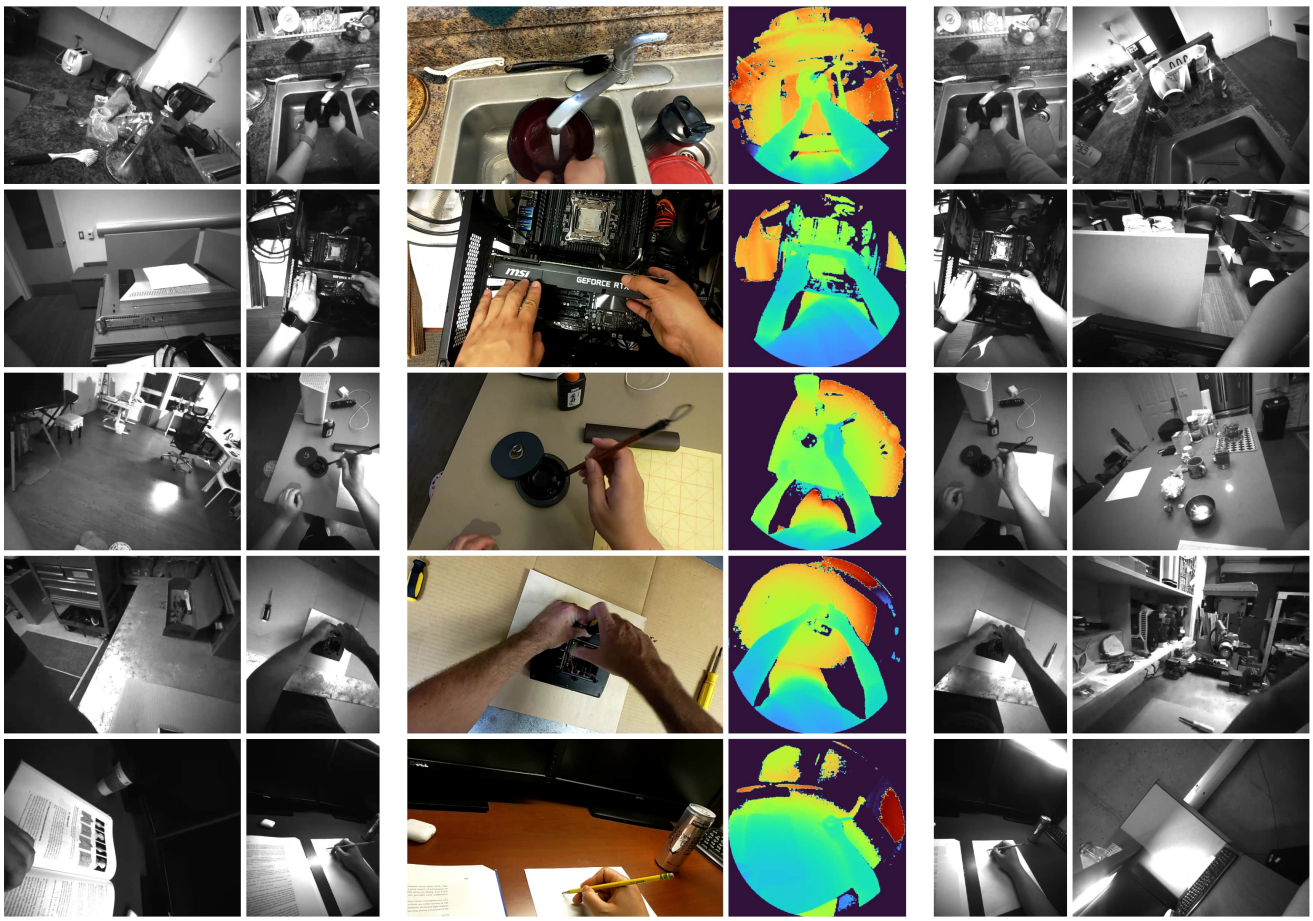

Figure 7. **2D visualizations of frames from raw video sequences (upper panel) and 3D visualizations of the capture environments (lower panel)**. The benchmark videos record camera wearers perform naturalistic tasks in real-world scenarios, such as cooking and repairing. Please refer to Figure 1 for the layout of each sensor on the HoloLens2.