Generating 6DoF Object Manipulation Trajectories from Action Description in Egocentric Vision

Supplementary Material

We provide detailed descriptions of our framework, dataset statistics, a baseline of Seq2Seq implementation, and implementation details of vision and point cloud-based language models. We also provide visualization of extracted trajectories and qualitative results of models in the supplementary video.

A. Dataset

A.1. Training Data

In addition to the descriptions in the main paper, we describe the details of temporal action localization, position sequence traction, and trajectory projection.

System Prompt for Temporal Action Localization. To obtain a manipulated object name, determine whether it is rigid, and localize the start and end timestamps of an action from a video clip, we use OpenAI GPT-4o [1] in two stages. First, to obtain the name of the manipulated object and determine whether the manipulated object is rigid or not, we conduct few-shot learning. We provide an action description and a system prompt containing several samples to GPT-4o without visual input. Figure 1 shows the system prompt and several samples for few-shot learning for this task. Second, we provide image frames assigned sequential indices, an action description, the manipulated object name, and a system prompt to GPT-4 to determine the start and end timestamps of the action for each video clip. The maximum sequence of image frames is eight, and embedded frames are depicted in Figure 2. Figure 3 shows the system prompt for this task.

Details of Position Sequence Extraction. To obtain segmentation maps of manipulated objects, we utilize the open-vocabulary segmentation model [4] as discussed in the main paper. However, egocentric videos of daily activities often involve the same objects within scenes, leading to incorrect object segmentation maps. To mitigate this issue, we employ a hand-object detection model [6] to detect interacted objects across all frames. We calculate the intersection-over-union (IoU) between the detected object bounding boxes and object segmentation map candidates, selecting the segmentation map with the highest IoU as the manipulated object segmentation map. Additionally, we filter out a video clip if the confidence score of the detection results falls below a threshold of 0.3.

Details of Trajectory Projection. To obtain projection matrices between image frames, we perform RANSAC-based global registration and colored iterative closest point (ICP) algorithm. For RANSAC-based global registration, we set the distance threshold to 0.03, the maximum number of iterations to 100,000, and the confidence level to 0.999. For the colored ICP algorithm, we set the distance threshold to 0.008, the maximum number of iterations to 100, the relative fitness to 1×10^{-6}, and the relative root mean square error to 1×10^{-6}.

	Music	Cooking	Bike Repair	Health Care
# Trajectories	816	20,529	1,448	5,704
# Frames	47,045	640,487	53,270	202,731
avg. Frames	58	31	38	36

Table 1. **Statistics of extracted trajectories in each scenario.** "avg." stands for average.

Details of Resource Dataset. To construct our dataset, we utilize the Ego-Exo4D [3] dataset, which encompasses eight diverse scenarios: dance, soccer, basketball, bouldering, music, cooking, bike repair, and health care. To focus on object interaction and ensure a stable trajectory projection process, we filter out the scenarios involving dance, soccer, basketball, and bouldering. Tab. 1 presents the number of trajectories and frames of trajectory for each scenario. The average number of frames of trajectory is calculated by dividing the total number of frames by the number of trajectories. As shown in Tab. 1, the cooking scenario constitutes the majority of our dataset. This result may be attributed to two reasons. First, the Ego-Exo4D dataset originally includes more instances of the cooking scenario compared to other scenarios. Second, objects in the bike repair and health care scenarios, such as a COVID-19 test plate, are more challenging to detect than those in the cooking scenario. Consequently, video clips from these scenarios are automatically filtered out. Moreover, the average number of frames per trajectory in the music scenario is higher than in other scenarios. This may be due to the characteristics of musical activities, playing musical instruments typically longer than other actions such as "grab a cup."

Failure Cases. There are unavoidable failure cases and we have carefully filtered out such cases through data curation methods. Fig. 4 illustrates failures from object segmentation and point cloud registration during the annotation process. Object segmentation can fail when multiple similar objects are present, while registration can fail when camera pose changes abruptly.

Data Curation Methods. In our study, we applied two filtering steps to remove inaccurate trajectories. First, we ex-

```
“System”: Based on the provided action description, answer an object that is actively being
manipulated (active object). Also, answer whether this active object is deformed during the
task.

“User”: “c slices the tomato with the black knife with right hand.”
“Assistant”: "active object: knife, rigid: true"

“User”: “c cut the paper with the scissors with his right hand.”
“Assistant”: "active object: scissors, rigid: false"

      ⋮

“User”: “c pours the water in the white ceramics bowl into the sink.”
“Assistant”: "active object: ceramics bowl, rigid: true"
```

Figure 1. **System prompt to obtain manipulated objects.**

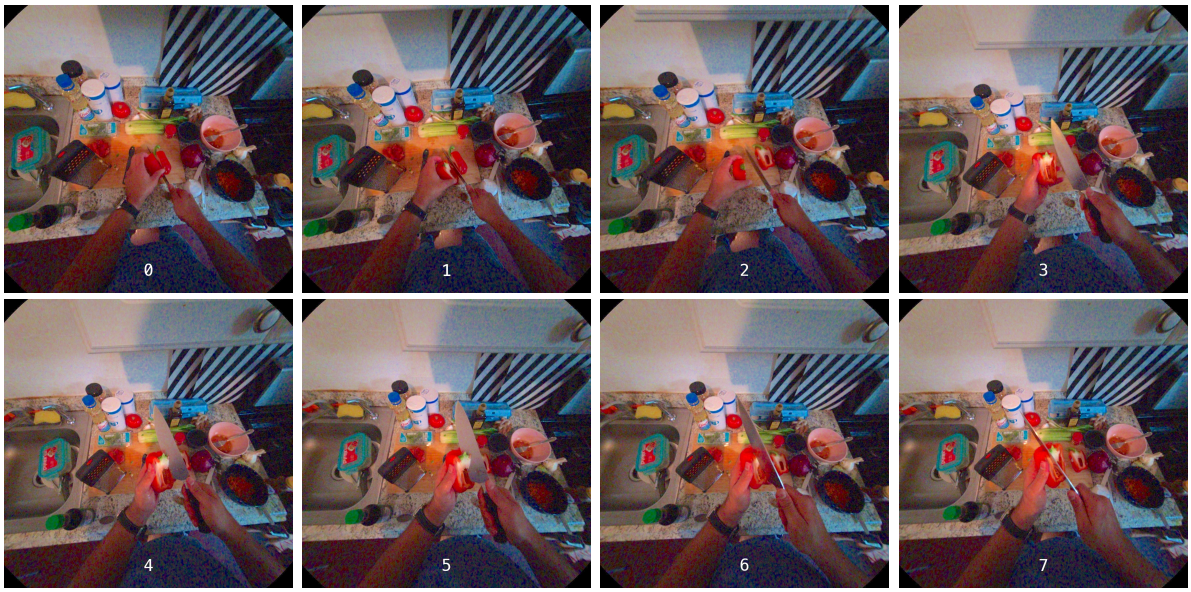


Figure 2. **Image frames embedded sequential indices for GPT-4o input.**

cluded incorrect segmentation results using a hand-object detector [6], such as Fig. 4 (a). Second, we removed trajectories that were out of frame in observation images, such as Fig. 4 (b).

A.2. Evaluation Data

In addition to the descriptions in the main paper, we describe how to determine manipulated objects within scenes, and how to annotate action descriptions using OpenAI GPT-4o [1].

Manipulated Objects Determination. To extract object manipulation trajectories, we need to detect which objects within a scene are being manipulated. To achieve this, unlike the approach used in constructing the training dataset, we utilize the annotated trajectories of each object provided in the HOT3D [2] dataset. For each video clip, we compute the displacement of each object and identify the object with the highest displacement as the manipulated object.

Action Description Generation. Since no textual information or action start and end timestamps exist in the HOT3D [2] dataset, we need to annotate them to align with our task setting. To achieve this, we adopt a similar workflow to the training dataset construction. We first split raw egocentric videos into several video clips, each spanning a four-second interval. Next, we perform temporal action localization and require the model to generate action descriptions. Additionally, we include an object name originally annotated in HOT3D as a user query to guide OpenAI GPT-4o in focusing on actions involving interaction with the object for each instance. Fig. 5 depicts the system prompt for this task.

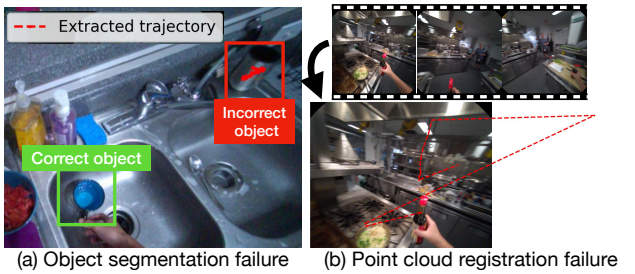
```
Identify the start frame and end frame in a sequence of frames extracted from a first-person
perspective video. Each frame is numbered.

### Definitions
- Interaction: The interaction occurs between a hand and the active object.
- Start frame: The frame where the described interaction begins.
- End frame: The frame where the described interaction ends.

### Hints
- The sequence of frames may contain irrelevant frames that only show hand movements or other
actions.
- Always ensure that the start frame number is less than the end frame number.

### Answer Format
Example: start frame: 5, end frame: 8
```

Figure 3. **System prompt for temporal action localization.**



(a) Object segmentation failure
"Pick the cup in the sink with his right hand."

(b) Point cloud registration failure
"Drop the wine bottle on the chopping board with his right hand."

Figure 4. **Failure cases.**

A.3. Dataset Statistics

Here, we provide detailed statistics for our dataset and the HOT3D evaluation dataset.

Vocabularies. Fig. 6 depicts word clouds of objects and verbs that appeared in action descriptions for both our dataset and HOT3D dataset. Our dataset consists of diverse objects for a wide range of verbs, enhancing models' generalization capability for read-world scenarios. Additionally, the objects in our dataset significantly differ from those in HOT3D, indicating that our approach successfully generates manipulation trajectories even for rare or unseen objects.

Average Displacement of Trajectories. Fig. 7 illustrates the statistics of average displacement of extracted object manipulation trajectories for both our dataset and HOT3D dataset. Although our dataset is constructed automatically, it has a similar distribution to that of HOT3D, thereby demonstrating the validity of our approach.

Variation in Each Element. Fig. 8 illustrates the distribution of each trajectory element for both our dataset and the HOT3D [2] dataset. The distribution of each element in our training dataset is similar to that of HOT3D. However, some variations in the rotational elements of our dataset slightly differ from those in HOT3D. These differences may arise from the domain disparity between Ego-Exo4D, the source of our dataset, and HOT3D. While the Ego-Exo4D dataset captures daily activities, HOT3D is designed for object tracking challenges. Consequently, the object manipulation motions in HOT3D involve actions with significant rotational movement, such as object inspection scenarios.

B. Baseline Seq2Seq Model

In our experiments, we utilize a Seq2Seq transformer with an MLP head as the baseline model [7]. The transformer comprises four layers, four attention heads, and a hidden size of 256. The MLP outputs a pose represented by $[x, y, z, \text{roll}, \text{pitch}, \text{yaw}]$. To address the issue of rotational continuity [8], we represent each angle $\theta \in \{\text{roll}, \text{pitch}, \text{yaw}\}$ using $[\cos(\theta), \sin(\theta)]$. After this transformation, each element of the parameters is normalized to the range $[0, 1]$.

C. Implementation Details of Our Model

We use AdamW [5] optimizer with a base learning rate of 2×10^{-5} for LLMs and 2×10^{-4} for other parameters across all backbone VLMs. We also use a linear warmup scheduler for 4 epochs on the EgoTraj. Models are trained for 30 epochs with a batch size of 8. After confirming that freezing LLMs leads to performance degradation, we unfreeze the LLMs during training. Additionally, we freeze all visual encoders during training.

References

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. 1, 2

[2] Prithviraj Banerjee, Sindi Shkodrani, Pierre Moulon, Shreyas Hampali, Fan Zhang, Jade Fountain, Edward Miller, Selen

```
Describe one main activity of a sequence of frames extracted from a first-person perspective
video. Each frame is numbered.
Besides, identify the start frame and end frame for the description. Each frame is numbered.

### Definitions
- Interaction: The interaction occurs between a hand and an object.
- Start frame: The frame where the described interaction begins.
- End frame: The frame where the described interaction ends.

### Hints
- The sequence of frames may contain irrelevant frames that only show hand movements or other
actions.
- Always ensure that the start frame number is less than the end frame number.
- Always follow one of each answer format.
- The subject of the description should be the camera-wearer: C.

### Answer Format
Description: c picks the knife on the table with the right hand.
start frame: 5
end frame: 8
```

Figure 5. **System prompt for HOT3D annotation.**

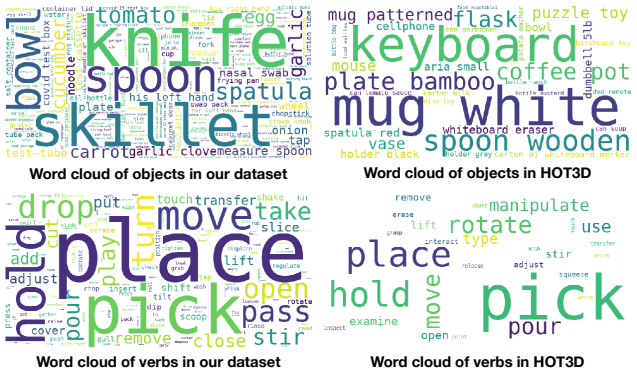
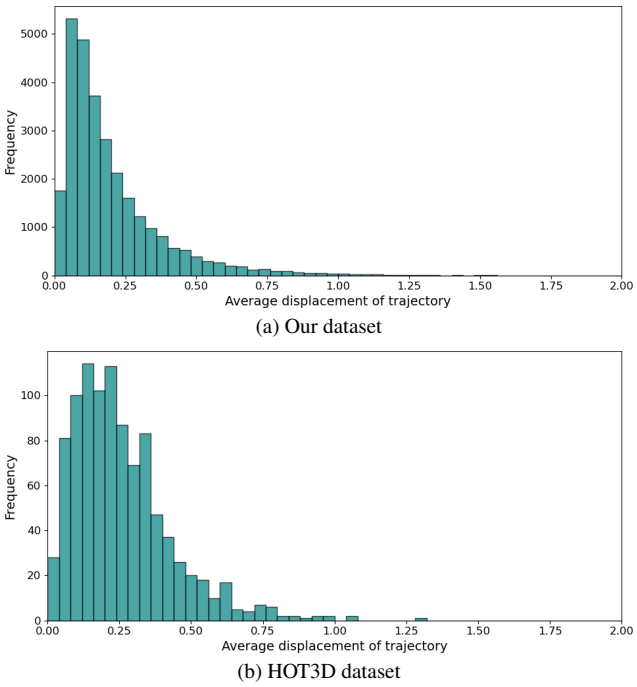


Figure 6. **Word cloud of objects and verbs in our dataset and HOT3D dataset [2].**



(a) Our dataset

(b) HOT3D dataset

Figure 7. **Distribution of the average displacement of trajectories in our dataset and HOT3D dataset.**

Basol, Richard Newcombe, Robert Wang, Jakob Julian Engel, and Tomas Hodan. Introducing hot3d: An egocentric dataset for 3d hand and object tracking. *arXiv preprint arXiv:2406.09598*, 2024. 2, 3, 4

[3] Kristen Grauman, Andrew Westbury, Lorenzo Torresani, Kris Kitani, Jitendra Malik, Triantafyllos Afouras, Kumar Ashutosh, Vijay Baiyya, Siddhant Bansal, Bikram Boote, et al. Ego-exo4d: Understanding skilled human activity from first-and third-person perspectives. *arXiv preprint arXiv:2311.18259*, 2023. 1

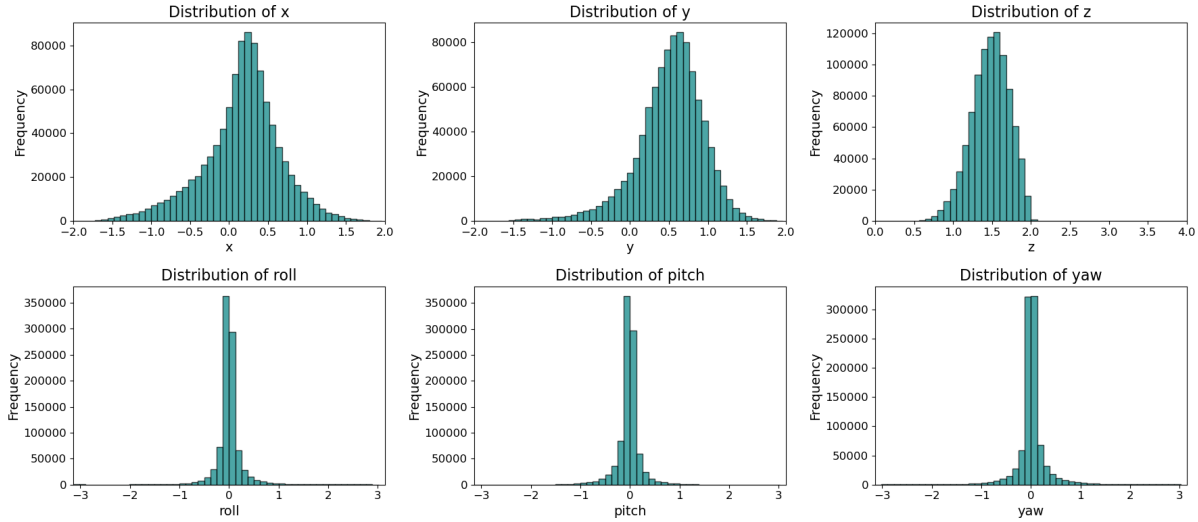
[4] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499*, 2023. 1

[5] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR*, 2019. 3

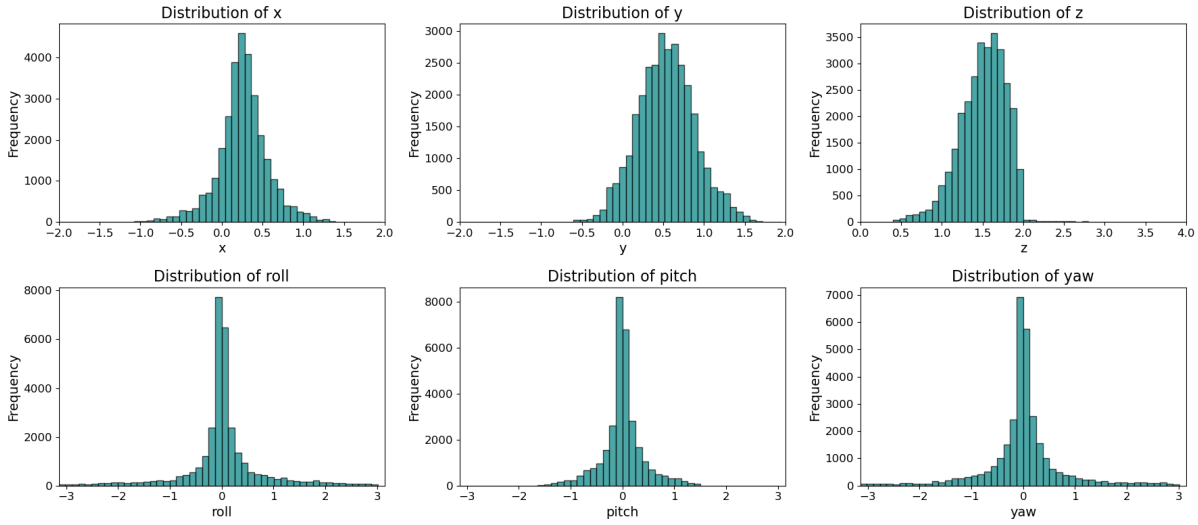
[6] Dandan Shan, Jiaqi Geng, Michelle Shu, and David F. Fouhey. Understanding human hands in contact at internet scale. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 1, 2

[7] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, 2017. 3

[8] Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. On the continuity of rotation representations in neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 3



(a) Our dataset



(b) HOT3D dataset

Figure 8. **Distribution of variations in each trajectory element for our dataset and HOT3D dataset.**