

802 A Implementation Details

803 A.1 Hyperparameters

804 We report important hyperparameters used for MEgoHand training in Table 1.

Table 1: Hyperparameters of MEgoHand Training.

Hyperparameter	Value
Prediction Trunk Size l	16
Integration Step Size δ	0.1
Gradient steps	50,000
Batch size	64
Learning Rate	3e-4
Optimizer	AdamW
Adam β_1	0.95
Adam β_2	0.999
Adam ϵ	1e-8
LR scheduler	cosine
Weight Decay	1e-5
Warmup Ratio	0.05
VLM text tokenizer	frozen
VLM vision encoder	unfrozen
DiT	unfrozen

805 A.2 Inverse MANO Pretraining

806 **Architecture.** The model architecture of the Inverse MANO Retargeting Network consists of
 807 PointNet encoder with 3-layer MLPs.

808 **Training Parameters.** We set $w_1 = 4.0$ and $w_2 = 5.0$ for \mathcal{L}_1 and \mathcal{L}_2 respectively. $\mathcal{L}_{\text{shape}}$ and $\mathcal{L}_{\text{recon}}$
 809 are both L1 loss, supervising shape feature β , translation t or rotation in 6D representation θ, r .

Visualization. Figure 1 shows using Inverse MANO Retargeting Network ϕ to label FPFA dataset.



Figure 1: We forward the MANO model to convert the outputs of Inverse MANO Retargeting Network ϕ to hand meshes, which are projected to the original frames in FPFA with the help of camera intrinsics and extrinsics.

A.3 Flow Matching

Recent work in high-resolution image and video synthesis has shown that flow matching can achieve strong empirical performance when combined with a simple linear-Gaussian (or optimal transport) probability path, given by:

$$q(\mathcal{H}_k^\tau | \mathcal{H}_k) = \mathcal{N}(\tau \mathcal{H}_k, (1 - \tau)^2 \mathbf{I}).$$

In practice, the network is trained by sampling random noise $\epsilon \sim \mathcal{N}(0, \mathbf{I})$, computing the "noisy actions" $\mathcal{H}_k^\tau = \tau \mathcal{H}_k + (1 - \tau)\epsilon$, and then training the network outputs $\nu_\theta(\mathcal{H}_k^\tau, h_k, z_k^{TDI})$ to match the denoising vector field:

$$\mathbf{u}(\mathcal{H}_k^\tau | \mathcal{H}_k) = \epsilon - \mathcal{H}_k.$$

The action expert uses a full bidirectional attention mask, so that all action tokens attend to each other. During training, we sample the flow matching timestep τ from a beta distribution that emphasizes lower (noisier) timesteps. At inference time, we generate actions by integrating the learned vector field from $\tau = 0$ to $\tau = 1$, starting with random noise $\mathcal{H}_k^0 \sim \mathcal{N}(0, \mathbf{I})$. We use the forward Euler integration rule:

$$\mathcal{H}_k^{\tau+\delta} = \mathcal{H}_k^\tau + \delta \nu_\theta(\mathcal{H}_k^\tau, h_k, z_k^{TDI}),$$

where δ is the integration step size. We use 10 integration steps (corresponding to $\delta = 0.1$) in our experiments. Note that inference can be implemented efficiently by caching the attention keys and values for the prefix h_k, z_k^{TDI} and only recomputing the suffix corresponding to the hand motion for each integration step.

A.4 Vision Language Model

Visual inputs are resized to 224×224 and encoded by SigLIP-2 with pixel shuffle [33], producing 64 spatially-aware visual tokens per frame, denoted as x^I . In parallel, textual instructions are processed by SmolLM2 to extract semantic representations x^T , facilitating cross-modal alignment.

A.5 HOI Datasets

Resources. We utilize a variety of publicly available egocentric hand-object interaction datasets in our experiments. Below is a brief description of each dataset along with its official website for reference:

- **H2O:** A large-scale egocentric dataset featuring hand-object interactions with both RGB and depth modalities. <https://taeinkwon.com/projects/h2o/>
- **HOI4D:** A dataset of human-object interactions, capturing fine-grained manipulation across various tasks. <https://hoi4d.github.io/>
- **HOT3D:** A dataset for hand-object tracking and manipulation with accurate annotations. <https://facebookresearch.github.io/hot3d/>
- **OAKINK2:** A comprehensive benchmark for large-scale egocentric manipulation with articulated object models. <https://oakink.net/v2/>
- **TACA:** A task-oriented dataset for contact-aware human-object interaction analysis. <https://taco2024.github.io/>
- **ARCTIC:** A richly annotated dataset for tracking hand-object contact and motion in egocentric scenarios. <https://arctic.is.tue.mpg.de/>
- **HOLO:** A large-scale dataset of household manipulation tasks captured in real-world environments. <https://holoassist.github.io/#HoloAssist>

Format. Our training corpora are built upon the LeRobot [5] dataset format, a widely used standard in the open-source robotics community and interaction learning community. Developed by Hugging Face, LeRobot is designed to make it easier to work with demonstration-based learning by offering a unified structure for storing, sharing, and utilizing demonstration data. Its popularity stems from its adaptability and the rich ecosystem of pretrained models and datasets available on the Hugging Face hub. The LeRobot format combines several well-established file types to ensure efficient storage and accessibility:

- **Tabular Data:** States, actions, and metadata are stored in Parquet files, which provide compact columnar storage and rapid access. This structure supports fast filtering and slicing—critical for training modern machine learning models.
- **Visual Data:** Observations in the form of videos (MP4) or image sequences (PNG) are referenced in the Parquet files, significantly reducing storage requirements while preserving accessibility.
- **Metadata:** Supplementary information such as dataset statistics and episode indexing is stored in JSON format, allowing structured, machine-readable access to dataset characteristics.

Demonstration sequences are organized into episodes, where each frame captures synchronized observations and corresponding actions. Observations typically include visual inputs (e.g., `observation.images.*`) and internal states (e.g., `observation.state`), while actions encode control directives. This episodic structure supports a wide range of learning paradigms. For imitation learning, the data enables supervised prediction of actions from observations. For reinforcement learning, it facilitates evaluation and optimization of decision-making strategies under varied state-action contexts. This standardized data format not only enhances reproducibility and interoperability across learning systems but also lowers the barrier to entry for researchers by providing a clean interface to high-quality interaction datasets.

While the LeRobot format provides a solid foundation, our work introduces several extensions to accommodate richer modality integration. We augment the standard format with the following components:

- **Modality Configuration:** A `modality.json` file is introduced within the `meta` directory to explicitly define the structure of the initial state and action vectors. This configuration maps each vector component to its semantic meaning and includes additional metadata relevant to each modality.
- **Fine-Grained Semantic Decomposition:** Departing from the monolithic vector approach of the original format, we decompose both state (initial hand state) and action (future hand motion trunk) vectors into semantically interpretable components—such as θ , β , r , and t —each annotated with its own data type, valid range, and transformation rules.
- **Multi-Annotation Integration:** The dataset format is extended to support multiple forms of annotations, such as task descriptions, validity indicators, and success labels. These annotations follow the LeRobot practice of storing indices in the Parquet files, with the corresponding content stored in auxiliary JSON files.
- **Rotation Representation Specification:** To ensure correct processing of rotational components during training, we require explicit declaration of the rotation representation used (e.g., quaternion, Euler angles, or axis-angle) for each relevant field.

These enhancements collectively enable more structured learning from complex demonstration data, with explicit modality definitions and robust support for multimodal supervision.

Preprocessing. For FPHA, we pretrain Inverse MANO Retargeting Network to label MANO parameters. For ARCTIC, HOT3D and OAKINK2, we adopt virtual RGB-D rendering to produce high-quality metric depth images in advance. All RGB and depth images are resized to 256×256 . It is worth noting that we split longer sequences to short clips (<500 steps) with the same task instruction for training and testing.

A.6 Computation Resources

MEgoHand is trained using 8×80GB NVIDIA A800 GPUs over approximately 24 hours. All evaluations and visualizations are performed on a single 80GB A800 GPU for around three hours.

A.7 Smooth Decoding

Decoding Strategy. As illustrated in Figure 2, at $t = 0$ MEgoHand receives initial hand MANO parameters, a egocentric RGB observation, and a depth map to predict trunk $t = 1 \dots l$. The predicted

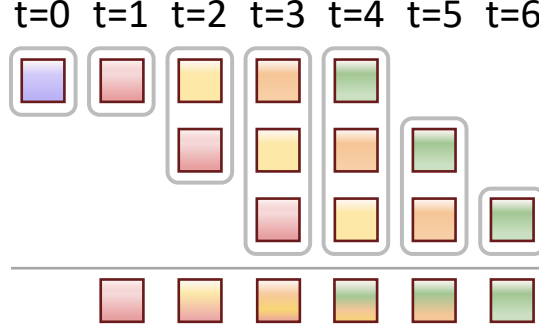


Figure 2: Illustration for smoothing predicted transformations.

905 wrist pose is relative to the initial hand pose and the predicted $\hat{\beta}$ is repeated from initial β . Then at
 906 $t = 1$, similarly, the predicted wrist pose $t = 2 \cdots l + 1$ is relative to the wrist pose predicted at $t = 1$,
 907 and so on. After converting all relative transformations to absolute transformations, we average all
 908 predictions at the same timestep to get smoother transformations.

909 **Visualization.** From Figure 3 we can see that smooth decoding strategy is effective in mitigating jitter.

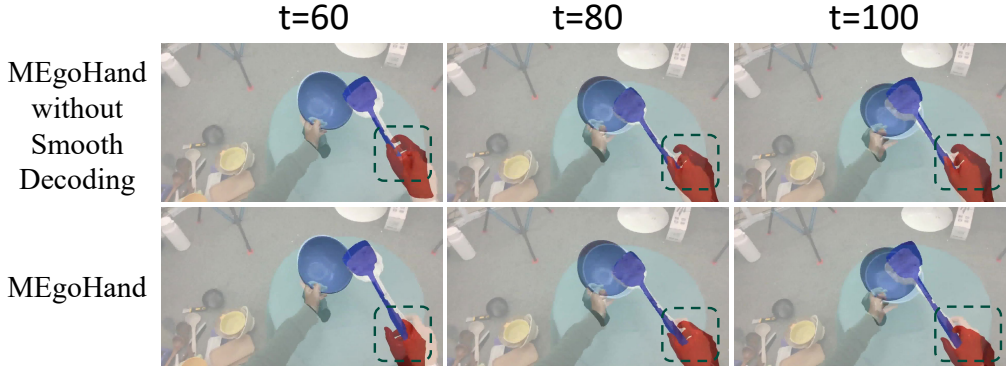


Figure 3: Frames randomly sampled from task "Stir the bowl with spatula" of TACO. Without decoding strategy, the predicted trajectory exhibits more fluctuations.

910

911 B Additional Visualizations

912 B.1 Zero-Shot Depth Estimation & Virtual Depth Rendering

913 In Figure 4, we visualize the zero-shot depth estimation of UniDepthV2 [29] and the virtual depth
 914 rendered from object models. Three datasets (OAKINK2,HOT3D,ARCTIC) are involved, as there
 915 are no real depth frames in these datasets.

916 B.2 HOI hand motion Generation

917 We visualize more clips of policy inference in Figures 5 and 6. MEgoHand is superior to baseline
 918 LatentAct in most cases.

919 C Empirical Results

920 We report the average metrics of MEgoHand in each dataset in Table 2.

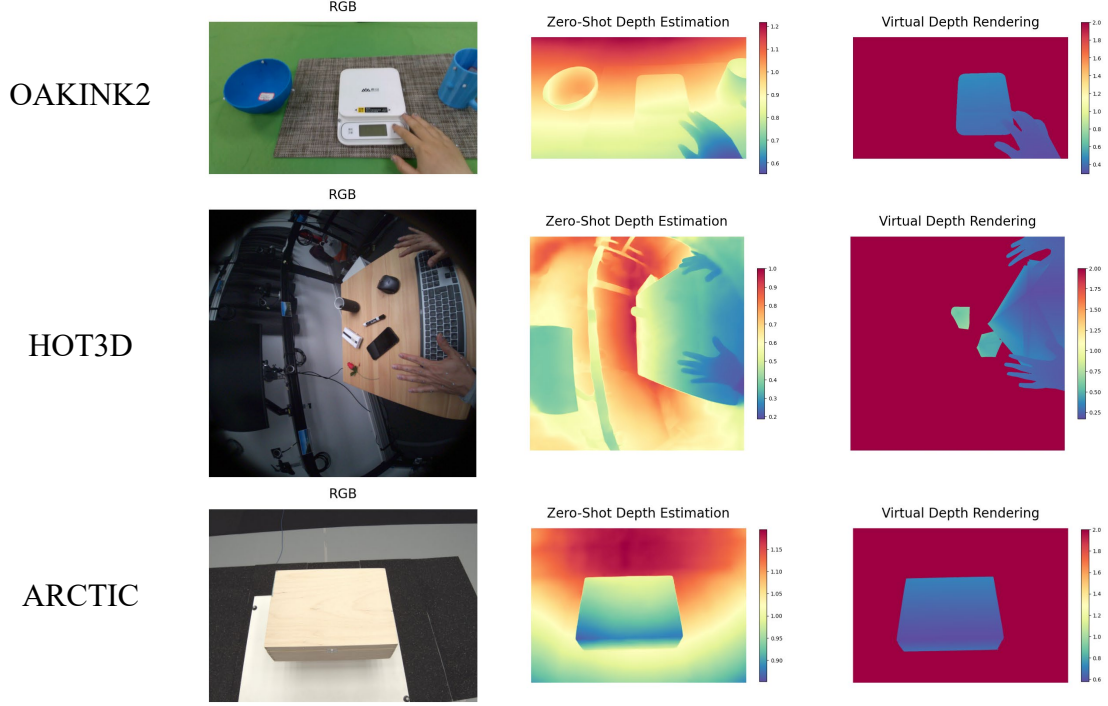


Figure 4: Colorbars indicate the absolute depth values (unit: m). The depth values of all depth frames fall within $[0, 2]$.

Table 2: Average metrics across evaluation (TACO, HOI4D, H2O, HOT3D, OakInk2) and testing datasets (ARCTIC, HOLO). The unit for MRE is radians; the remaining metrics are measured in centimeters.

Dataset	MPJPE	MPJPE-PA	MPVE	MPVE-PA	MWTE	MRE
H2O	3.013	0.352	2.969	0.334	2.450	0.099
HOI4D	8.958	0.856	8.933	0.826	8.462	0.213
HOT3D	6.437	0.236	6.352	0.228	5.045	0.086
OAKINK2	3.424	0.217	3.380	0.205	2.837	0.071
TACO	4.936	0.358	4.899	0.346	4.465	0.131
ARCTIC	7.358	1.161	7.268	1.106	5.958	0.398
HOLO	5.775	0.697	5.747	0.673	5.437	0.271

D Social Impact

MEgoHand forwards an important step toward universal hand-object motion generation from multiple modalities including task instruction, RGB observation, depth image, and initial conditions. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

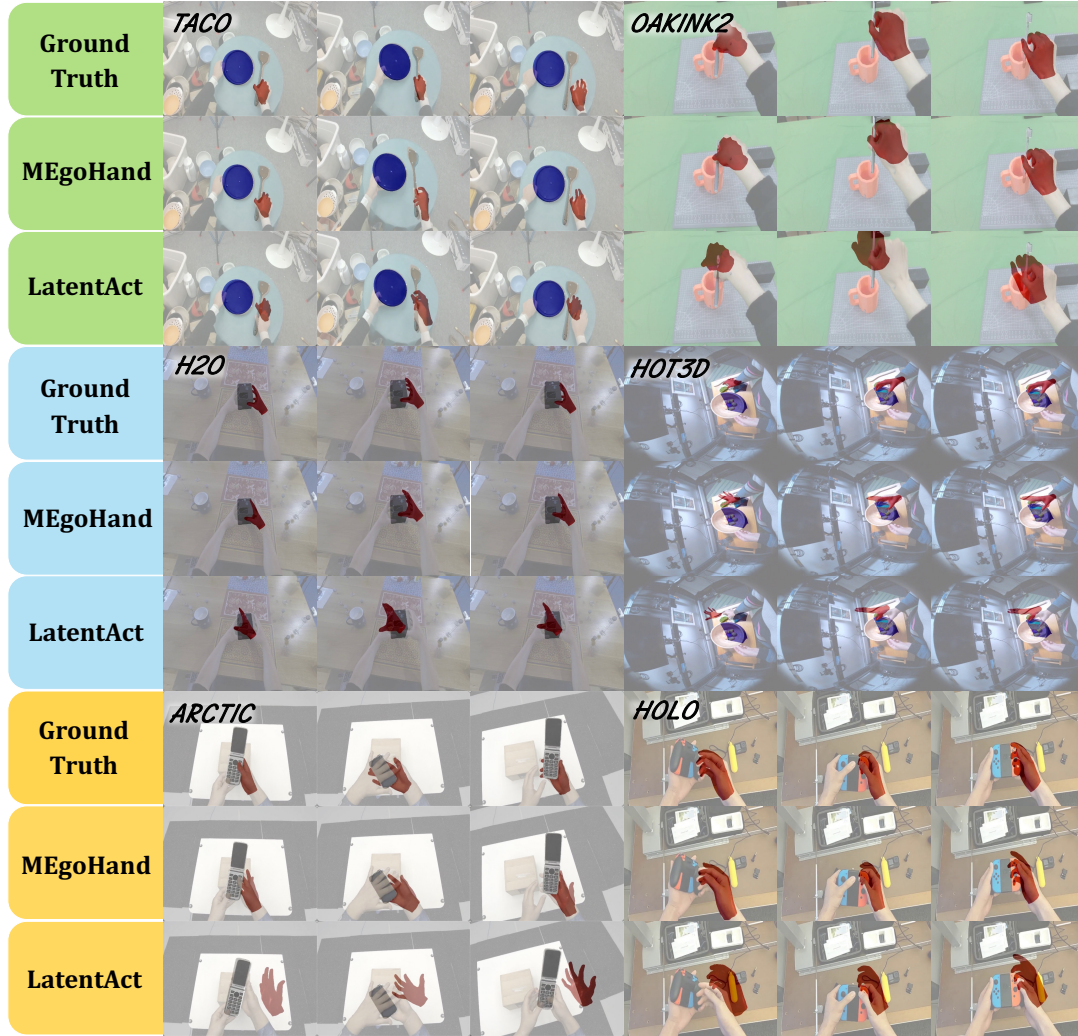


Figure 5: Additional visualizations of LatentAct and MEgoHand. Green part is sampled from training sets. Blue part is sampled from evaluation sets. The Yellow part is sampled from testing sets.

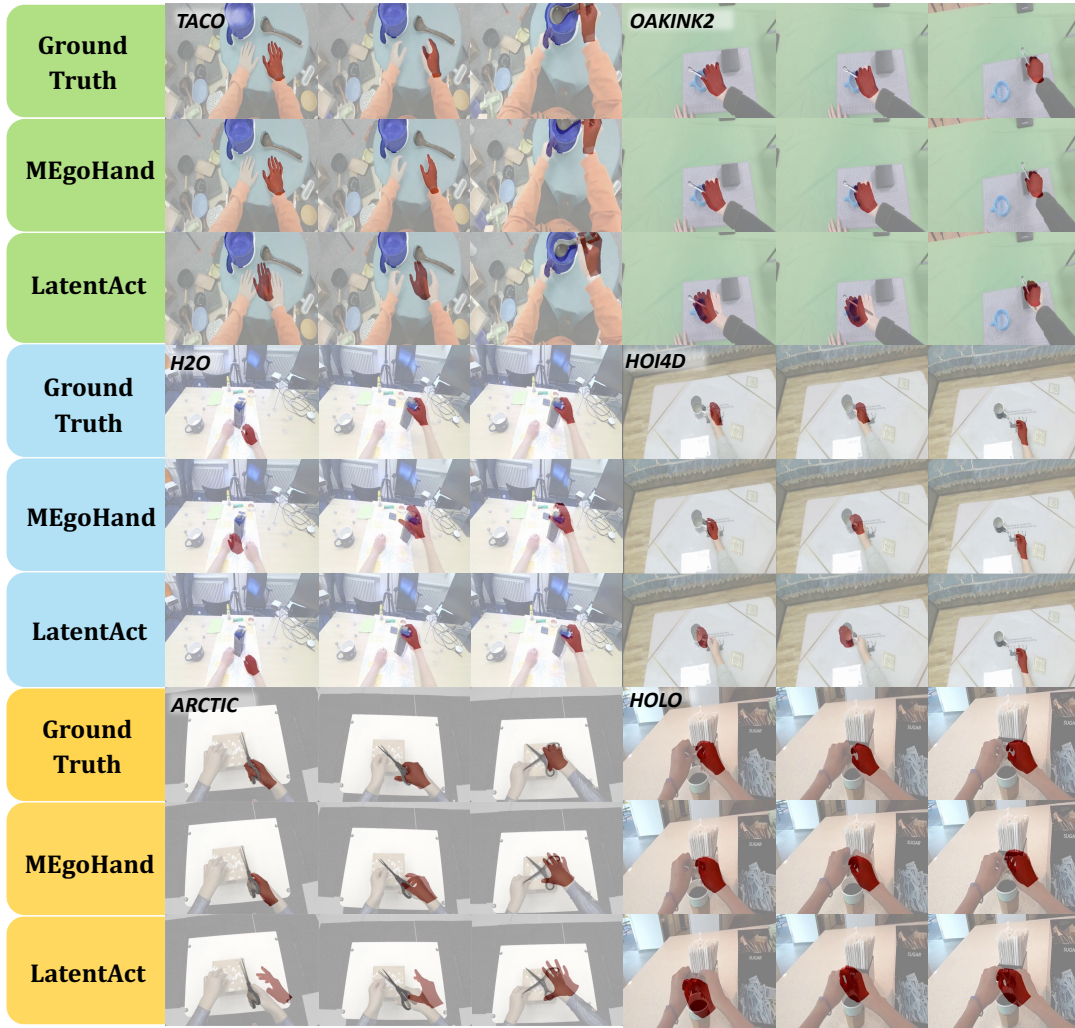


Figure 6: Additional visualizations of LatentAct and MEgoHand. Green part is sampled from training sets. Blue part is sampled from evaluation sets. The Yellow part is sampled from testing sets.