HERMES: Human-to-Robot Embodied Learning from Multi-SouRce Motion Data for MobilE DexterouS Manipulation

Zhecheng Yuan[1,2*], **Tianming Wei**[1,2*], **Langzhe Gu**[1,2*], **Pu Hua**[1,2], **Tianhai Liang**[1,2], **Yunapei Chen**[3], **Huazhe Xu**[1,2]
[1] Tsinghua University, [2] Shanghai Qi Zhi Institute, [3] Peking University

Abstract: Leveraging human motion data to impart robots with versatile manipulation skills has emerged as a promising paradigm in robotic manipulation. Nevertheless, translating multi-source human hand motions into feasible robot behaviors remains challenging, particularly for robots equipped with multi-fingered dexterous hands characterized by complex, high-dimensional action spaces. In this paper, we introduce HERMES, a human-to-robot learning framework for mobile bimanual dexterous manipulation. First, HERMES formulates a unified reinforcement learning approach capable of seamlessly transforming heterogeneous human hand motions from multiple sources into physically plausible robotic behaviors. Subsequently, to mitigate the sim2real gap, we devise an end-to-end, depth image-based sim2real transfer method for improved generalization to real-world scenarios. Furthermore, to enable autonomous operation in varied and unstructured environments, we augment the navigation foundation model with a closed-loop Perspective-n-Point (PnP) localization mechanism, ensuring precise alignment of visual goals and effectively bridging autonomous navigation and dexterous manipulation. Extensive experimental results demonstrate that HERMES consistently exhibits generalizable behaviors across diverse, in-the-wild scenarios, successfully performing numerous complex mobile bimanual dexterous manipulation tasks. Project Page https://hermes-manipulation.github.io/

Keywords: Mobile bimanual dexterous manipulation, Sim2real, Learning from human motion.

1 Introduction

Achieving human-level dexterity for robots has long been a central challenge in robotic research. The prospect of bimanual robotic systems with dexterous hands that mirror human physiology holds the promise of seamlessly integrating robots into daily human activities and environments. Despite notable progress, how to capitalize on the abundance of available human data and develop algorithms suited to intricate and high-precision dexterous manipulation remains underexplored.

Humans continuously generate diverse bimanual manipulation data, inherently serving as natural guidance for robots to emulate human-like behaviors. Several previous studies [1, 2, 3, 4, 5] have attempted to extract trajectories of human hands and manipulated objects from video data, subsequently applying them to robotic manipulation tasks. Nevertheless, these methods have predominantly targeted robots equipped with simple gripper-based end effectors, failing to generalize effectively to dexterous hands due to the vastly greater complexity of action space. Despite recent advances that utilize kinematic retargeting approaches to produce human-like robotic motions [6, 7, 8, 9, 10], these approaches still fall short in achieving physically-aware pose retargeting and bridging the embodiment gap to derive feasible robot actions capable of successfully accom-

*Equal Contribution

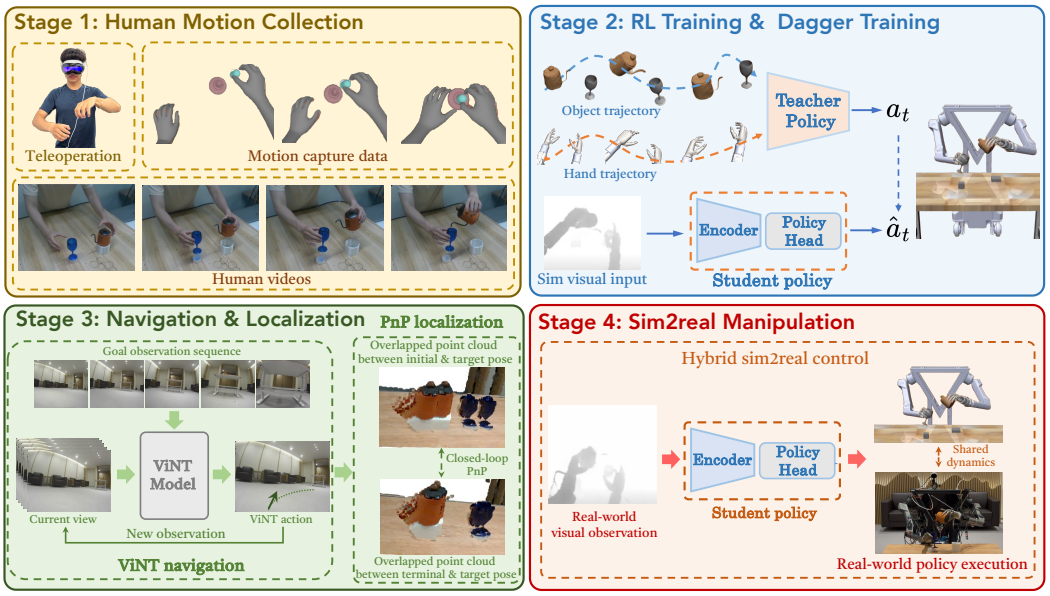

Figure 1: **The main pipeline of HERMES.** HERMES comprises a four-stage pipeline for achieving mobile bimanual dexterous manipulation through sim2real transfer. First, we acquire a one-shot human demonstration drawn from diverse sources. Then, in stage 2, we train a state-based RL teacher policy, then apply DAgger to distill into a vision-based student policy. Following this, HERMES execute long-horizon navigation using ViNT, followed by real-time PnP to finely adjust the robot's pose and achieve precise alignment in stage 3. Once localization is achieved, the student policy is deployed in a zero-shot fashion directly in the real world.

plishing the intended tasks. ***A critical limitation lies in the omission of explicit modeling of interactions between robotic hands and manipulated objects, a fundamental component of manipulation tasks.*** Consequently, neglecting these interactions undermines the robot's ability to fully understand and adapt to the dynamics of manipulation scenarios.

Therefore, in an attempt to address the aforementioned challenges, recent approaches have begun leveraging Reinforcement Learning (RL) paradigms [11, 12, 13], allowing robots to autonomously explore feasible motion strategies under the guidance of kinematic reference trajectories. These methods commonly design general reward functions encompassing object tracking, hand configurations, and collision dynamics. Maximizing such rewards drives the robot toward successful execution of complex manipulation tasks. Nonetheless, existing works [12, 13] typically draw on limited human motion data sources and some have not transferred the trained robot behaviors to the physical world. Such limitations not only hinder the evaluation of whether the learned policies exhibit behaviorally plausible performance in the real world, but also preclude the integration of sim2real methodologies necessary for enabling closed-loop policy control and deploying in various environmental conditions. Motivated by these challenges, we propose HERMES, a versatile human-to-robot embodied learning framework tailored for mobile bimanual dexterous hand manipulation. HERMES offers the following three advantages: **1. Diverse sources of human motion**: Our framework supports several human motion sources, including teleoperated simulation data, motion capture (mocap) data, and raw human videos. We also provide corresponding approaches for data acquisition, enabling HERMES to efficiently transform varied human motion data into robot-feasible behaviors through RL. Furthermore, these tasks share a uniform set of reward terms, obviating the necessity of designing intricate and task-specific reward functions. In contrast to the methods that depend on collecting a large amount of demonstrations, we can achieve generalizable policy by editing a single reference human motion trajectory coupling with RL training. **2. End-to-end vision-based sim2real transfer**: HERMES facilitates robust vision-based sim2real transfer by employing DAgger distillation, which converts state-based expert policies into vision-based student policies. Moreover, we introduce a generalized, object-centric depth image augmentation and hybrid control approach, effectively bridging the perception and dynamic sim2real gap. **3. Mobile**

manipulation capability: Our method endows robots with mobile manipulation skills. Building upon ViNT [14], we develop a RGB-D based module for precise localization wherein the task is modeled as a Perspective-n-Point (PnP) problem and addressed through an iterative process. This ensures seamless integration with subsequent manipulation tasks and unlock the policy's capacity to operate autonomously across a broad spectrum of real-world environments.

2 Method

2.1 Collect One-shot Human Motion

To validate the effectiveness and robustness of HERMES, we employ three distinct sources of human motion: teleoperation in simulation, motion capture data obtained from public datasets, and hand-object poses extracted from raw videos. Moreover, by leveraging merely a single human reference trajectory in conjunction with RL training, we are able to derive the generalizable robot policy without the need for collecting extensive demonstrations.

Teleoperation in simulation: We provide access to the pre-configured simulation that enables direct teleoperation of the robot for collecting demonstrations. The Apple Vision Pro is utilized to extract hand poses and arm movements, with data captured at a frequency of 75 Hz.

Mocap data: In contrast to direct teleoperation in simulation, retargeting mocap data to robotic hands presents significant challenges due to the embodiment gap between human and robotic hand structures. This discrepancy renders the retargeted trajectories from mocap data unsuitable for direct replay in simulation. Consequently, RL is often employed to enable robots to learn the desired behaviors from reference trajectories. In our study, we utilize the OakInk2 mocap dataset [15] to acquire human motion data for this purpose.

Extracted arm and hand poses from videos: Leveraging video data holds considerable promise for unlocking vast quantities of information to facilitate robot learning. To this end, we also provide a pipeline for extracting human hand poses and object trajectories directly from raw video. To acquire the hand poses, we first employ WiLoR [16] to detect the hands in each video frame and extract 2D hand keypoints along with their corresponding 3D counterparts. We then select a relatively stable subset of keypoints for the subsequent estimation, specifically those located at the wrist and the metacarpophalangeal joints. The spatial translation of the wrist in the camera coordinate system is estimated by solving a Perspective-n-Point (PnP) problem [17] based on the 2D-3D correspondences, while the palm's orientation is derived by fitting a plane to the selected 3D keypoints. Regarding the manipulated objects, we employ FoundationPose [18] to estimate the object poses directly from video frames, and utilize ARCode [19] scanning to reconstruct the object mesh. By leveraging the aforementioned procedures, we can align the hand and object poses extracted from the video with the robot's frame to facilitate the subsequent learning process.

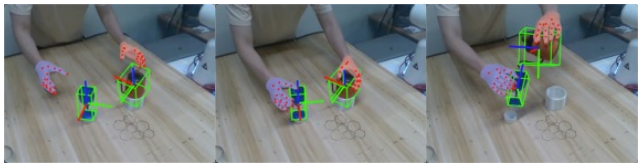

Figure 2: **Pose extraction from videos.**

Synthesize multiple trajectories: To obtain a more generalizable policy, we perform the trajectory editing for the one-shot human motion reference by randomizing the object's position and orientation in a predefined range. The hand and object poses across the augmented trajectories are transformed as follows:

$$\hat{\mathbf{A}}^{\text{pose}}\left[\tau_k\right] = \mathbf{T}^{\text{trans}} \cdot \mathbf{A}^{\text{pose}}\left[\tau_k\right]. \tag{1}$$

For any given frame k in the trajectory τ, we apply a transformation matrix $\mathbf{T}^{\text{trans}}$ to alter its pose, where $\mathbf{A}^{\text{pose}}$ may represent either the object pose or the hand pose. By editing the reference trajectory, we enable spatial generalization from a single human motion demonstration, obviating the need to manually collect large numbers of teleoped demonstrations.

Upon obtaining synthesized object and hand trajectories from various data sources, we initially employ the DexPilot retargeting method [20] to map the captured human hand poses onto corresponding robot hand configurations. Subsequently, reinforcement learning is leveraged to refine and adapt the initialized robot behaviors.

2.2 Generalizable Reward Design for Manipulation

Standard reinforcement learning typically relies on hand-crafted reward functions tailored to each specific task. However, designing such complicated reward structures often impedes scalability and usability, particularly for the dexterous hand. To alleviate this issue, we leverage one-shot human demonstration combined with a generalizable reward formulation, enabling the reuse of a unified reward function across tasks and facilitating the straightforward construction of challenging, long-horizon manipulation tasks. Specifically, we design the following three reward terms:

Object-centric Distance chain: Capturing the dynamic spatial relationships between the human hands and the object stands as a pivotal factor in enabling the policy to acquire fine-grained hand-object interaction skills. We designate the coordinates of the fingertips and palm of the hand, along with the center of the object's collision mesh, as keypoints. By modeling the temporal evolution of vectors between these keypoints, we formulate the following reward function:

$$r_{\text{chain}} = \begin{cases} \exp\left\{\frac{1}{n}\sum_{i=1}^{n}\left\|\vec{r}_{\text{ref}}^{(i)} - \vec{r}^{(i)}\right\|\right\}, & \text{if } N_{\text{contact}} \geq N_{\text{num}} \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

where $\vec{r}^{(i)}$ is the vector from object center to the fingertip or palm. Furthermore, we incorporate contact information into this reward term. Specifically, during the computation of the distance chain, we also evaluate the number of contact points between the fingertips and palms of both hand mesh $\mathbf{C}_{\text{hand}}$ and the object's collision mesh $\mathbf{C}_{\text{obj}}$. This reward component is activated only when the number of contact points N_{contact} exceeds a predefined threshold N_{num}, ensuring that the policy attends to physically meaningful hand-object interactions.

Object trajectory tracking: For manipulation tasks which adopt human motion, a critical indicator of policy success lies in its ability to track and follow the desired object trajectory. To this end, we introduce an additional reward component that explicitly aligns the policy's behavior with the target object's trajectory:

$$r_{\text{obj}} = \exp\left(-k_1 \cdot \|\mathbf{p}_{\text{obj}} - \mathbf{p}_{\text{ref}}\|^2 - k_2 \cdot (d_{\text{quat}}(\mathbf{q}_{\text{obj}}, \mathbf{q}_{\text{ref}}))^2\right) \tag{3}$$

where k_1, k_2 are the coefficients corresponding to the position and orientation terms in the r_{obj}. $\mathbf{p}_{\text{obj}}$ and $\mathbf{q}_{\text{obj}}$ represent the current position and orientation of the object, while $\mathbf{p}_{\text{ref}}$ and $\mathbf{q}_{\text{ref}}$ denote the position and orientation along the reference trajectory. The term d_{quat} measures the distance between two quaternions. We also incorporate a power-penalty term to enhance the smoothness of policy execution and to alleviate the jittering actions. By integrating all these reward components, the policy is endowed with the capacity to tackle a wide spectrum of challenging and diverse manipulation tasks. We adopt DrM [21], an off-policy method, leverages a dormant ratio mechanism [22] to enhance exploration capabilities and demonstrates high sample efficiency.

3 Sim-to-real Transfer

The training of state-based RL policies typically relies on privileged information which is not accessible in real-world deployment scenarios. Consequently, it is imperative to distill the state-based policy into a visual policy for achieving sim2real transfer.

Leveraging depth image as visual input: Prior work has explored the use of depth images for vision-based sim2real transfer. However, they often necessitate intricate and highly customized

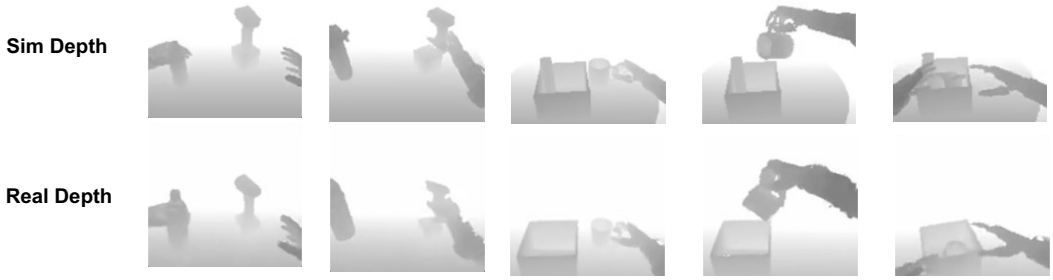

Figure 3: **Depth image visualization.** We present a visual comparison between simulated and real-world depth maps across two different tasks. Notably, after applying our preprocessing pipeline, the depth representations of the hand and object exhibit a strong semantic correspondence, highlighting the efficacy of HERMES in bridging the sim2real gap.

augmentation strategies to bridge the gap. In this work, we introduce a more versatile, manipulation-tailored egocentric depth-image augmentation method. Specifically, we clip depth values beyond a threshold distance $\mathbf{d}$ (set per task). For real depth images, missing depth values resulting from edge capture failures are filled in with the maximum depth. To emulate real-world edge noise and blur in simulation, we augment simulated depth images by adding Gaussian noise and Gaussian blur during training. Additionally, to mimic missing depth values, we randomly set 0.5% of pixel values in simulation-rendered images to the maximum depth. As illustrated in Figure 3, our augmentation not only semantically aligns simulated renderings with real-world depth images, but also preserves crucial depth disparity cues essential for accurate visuomotor control.

3.1 DAgger Distillation Training

In DAgger training, the state-based expert policy acts as the teacher to guide the learning of a visual student policy. In contrast to prior approaches that distill to object masks or segmented images, HERMES directly distills the state into raw visual observations of entire visual scenarios. This design obviates the need for explicit camera calibration and facilitates the acquisition of the robot's in-the-wild generalization capabilities. Furthermore, we introduce a series of auxiliary design choices aimed at enhancing both the asymptotic performance of DAgger training.

Hybrid Sim2real Control: To mitigate the gap between simulation and real-world dynamics as well as proprioceptive information, we adopt a hybrid control strategy: real-world visual observations are used to infer the actual action, which is then applied to the simulation environment to perform a forward step. The updated joint positions of the simulated robot are subsequently transferred to the real robot for execution. By sharing the same Inverse Kinematics (IK) method and dynamic parameters across simulation and the real world, this approach not only enables the policy to adapt its behavior based on real-world environmental variations but also effectively narrows the sim2real discrepancy. The pipeline is shown in Figure 4.

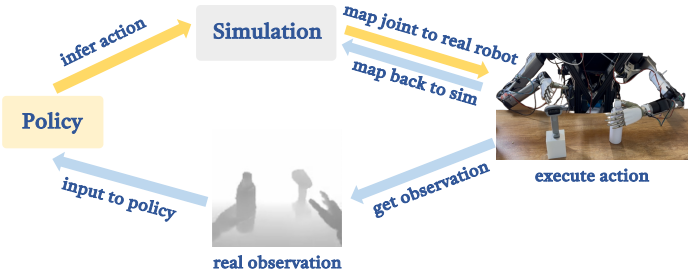

Figure 4: **Hybrid Sim2real Control.**

4 Navigation Methodology

To endow the trained visuomotor policy with navigation capabilities, HERMES integrates an image-goal navigation foundation model [14, 23, 24] that operates solely on RGB inputs and supports

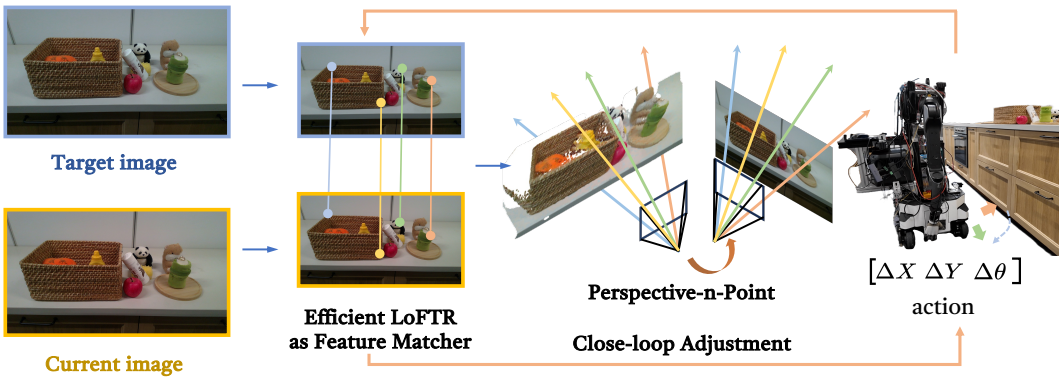

Figure 5: **The pipeline of closed-loop PnP localization.** We first employ the efficient LoFTR to extract dense visual features, followed by estimating the transformation between the current frame and the goal location via solving the PnP problem. Subsequently, we use PID controller to execute the action. This entire process is executed in a closed-loop manner and continues iteratively until the spatial discrepancy between the robot's current pose and the goal falls below a predefined threshold.

long-horizon, in-the-wild navigation. This framework allows for a seamless and low-cost fusion of manipulation and navigation modules, without necessitating additional fine-tuning of either component. We choose ViNT [14] for achieving image-goal robotic navigation. We deploy ViNT on our customized robotic system, operating at a frequency of 7.6 Hz. The action space is the relative waypoints so that ViNT can own the cross-embodiment property. ViNT not only enables long-range, in-the-wild navigation but also demonstrates effective zero-shot generalization capability without necessitating model fine-tuning.

For our mobile manipulation tasks, even moderate discrepancies between the robot's final pose and the target pose can lead to the manipulation policy failing to finish the task. However, ViNT does not guarantee termination within a sufficiently tight error bound. To address this, we introduce a local refinement step after ViNT completes navigation: a closed-loop Perspective-n-Point (PnP) localization algorithm is employed to adjust the robot's pose, ensuring closer alignment with the goal image pose.

As shown in Figure 5, we first utilize the neural feature matching module Efficient LoFTR [25] to detect the correspondence between the current robot captured image I_c and the goal image I_g. Then the detected features are lifted to 3D space with respect to the robot's current coordinate frame by leveraging the camera intrinsic matrix and the depth map. Next, we leverage the RANSAC PnP [26] and refine PnP algorithm [27, 28] to compute the relative rotation and translation between the robot's current viewpoint and the goal pose that can minimize the reprojection error. By leveraging real-time feedback from PnP as the robot incrementally converges toward the target pose, we are able to iteratively refine the pose estimation, thereby attaining more accurate visual correspondence. After getting the target pose calculated by our closed-loop PnP localization algorithm, we utilize a Proportional-Integral-Derivative (PID) controller [29] to adjust the pose of our robot. The input of the controller is the instantaneous position and orientation error between the robot's desired state and its actual state.

5 Experiments

In this section, we perform an extensive series of experiments aimed at evaluating the capabilities of HERMES across various aspects, including navigation and manipulation. Specifically, our primary experiments are designed to: (1) verify the efficacy of HERMES in efficiently and robustly transforming diverse human motion data into robot-plausible behaviors; (2) exhibit the effectiveness of our method in sim2real transfer; (3) quantify the accuracy and reliability of our navigation localization approach; (4) demonstrate the effectiveness of HERMES in mobile manipulation.

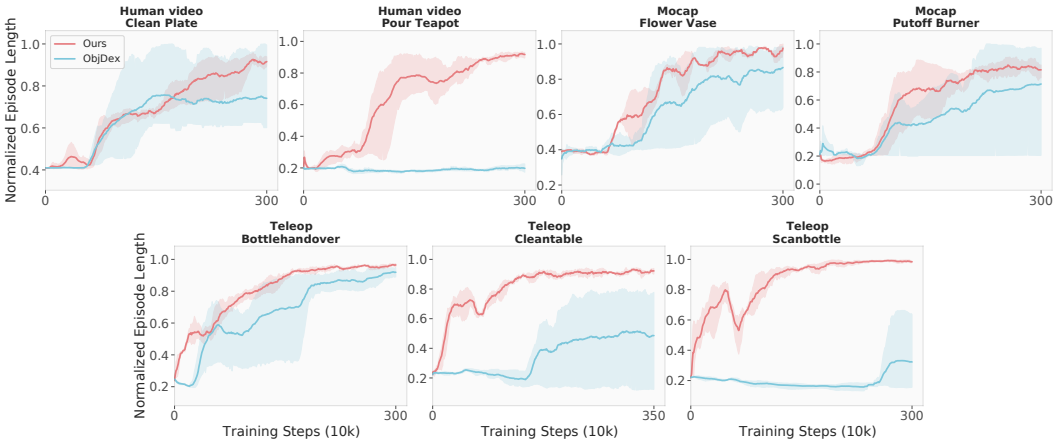

Figure 6: **The training curve of HERMES.** The horizontal axis denotes the training steps, while the vertical axis represents the normalized task length successfully accomplished by the policy. *Teleop* refers to one-shot human motion teleoperation in simulation, *Human video* denotes trajectories extracted from video data, and *Mocap* corresponds to motion derived from mocap datasets. All results are evaluated across 3 seeds.

Table 1: Real-world manipulation evaluation results. Across 6 real-world bimanual dexterous manipulation tasks, HERMES obtains $+\mathbf{54.5}\%$ performance gains on average.

Method \ Tasks	bottlehandover	cleantable	scanbottle	putoffburner	cleanplate	pourteapot	**Average**
HERMES	**66.7**	**60.0**	**73.3**	**66.7**	**66.7**	**73.3**	**67.8**$\pm$**5.0**
raw depth	6.7	0.0	0.0	20.0	13.3	40.0	13.3$\pm$15.2

5.1 Sample Efficiency of HERMES

We evaluate the training sample efficiency of HERMES across seven tasks. For each task, the source of the one-shot human motion demonstration is indicated in the title of each sub-figure in Figure 6. The vertical axis in the figure represents the proportion of the trajectory length successfully executed by the current policy relative to the total length of the trajectory. As demonstrated in Figure 6, regardless of the origin of the human motion data, HERMES reliably succeeds in converting human hand and arm actions into generalizable robot-executable behaviors.

Additionally, we compare training performances with ObjDex [11]. ObjDex defines its reward based on the tracking of the object's joint movement, translations, and orientations. We re-implement this reward formulation within our own algorithmic framework. Figure 6 indicates that HERMES exhibits superior performance relative to ObjDex across all tasks. In tasks such as *Bottle Handover*, *Flower Vase*, and *Putoff Burner*, where interactions involve only a single object, ObjDex is able to complete the tasks; however, HERMES can achieve higher sample efficiency during training. Furthermore, in more intricate tasks involving multi-object interactions, ObjDex consistently fails, irrespective of the type of human motion data provided. Contributed to our object-centric distance chain, HERMES is capable of robustly acquiring diverse manipulation skills even in long-horizon, multi-object environments. Moreover, HERMES demonstrates high sample efficiency and successfully learns policies in 3M training steps.

5.2 Real-world Manipulation Evaluation

After conducting DAgger training, we subsequently transfer the trained visual student policy to the real world in a zero-shot manner for most tasks. It should be noted that for the tasks *pour teapot* and *putoff burner*, the presence of substantial noise in the trajectory or transparent objects leads to excessively jittering motions, along with discrepancies between simulated and real-world object shapes. Consequently, we additionally fine-tune the policy using 5 extra real-world trajectories collected via policy rollouts. Table 1 presents the generalization performance of the policy evaluated across

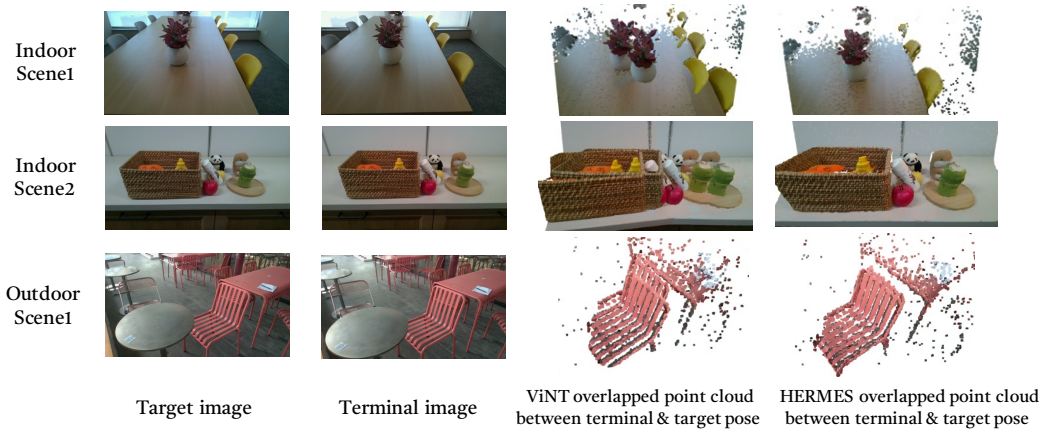

Figure 7: **The visualization of navigation results.** The left two columns depict a comparison between the target image and the terminal image achieved by our method. The right two columns present the point clouds captured at the end of navigation by ViNT and HERMES, compared against the point cloud of the target position.

different object placements and poses, with each task assessed over 15 trials. HERMES not only successfully achieves zero-shot transfer for diverse long-horizon or contact-rich bimanual dexterous manipulation tasks, but also surpasses the baseline by $+\mathbf{54.4}\%$ in success rate. These experimental results substantiate HERMES's capability to effectively bridge both visual and dynamic gaps, enabling successful sim2real transfer and demonstrating intricate manipulation skills. Moreover, for the two tasks involving fine-tuning with real-world rollout trajectories, owing to the reduced visual discrepancy achieved by HERMES, the trained policy exhibits enhanced generalization capabilities compared to the raw depth baseline.

5.3 Mobile Manipulation Evaluation

To evaluate the mobile manipulation ability of HERMES, we integrate the entire pipeline across all tasks. Each trained policy is tested over 10 runs. As illustrated in Figure 8, HERMES demonstrates strong real-world navigation, precise localization, and dexterous manipulation capabilities. We also apply the identical manipulation policy equipped with ViNT as a baseline. Figure 8 reveals that, without closed-loop PnP localization, the policy cannot generalize or successfully complete tasks when faced with significant positional and rotational shifts. Conversely, HERMES achieves a notable $+\mathbf{54.0}\%$ improvement in manipulation success rate compared to pure ViNT. These findings underscore that closed-loop PnP localization is the essential bridge linking navigation and manipulation, enabling both modules to synergize for enhanced performance.

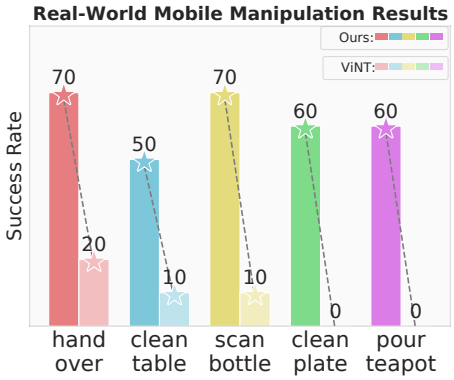

Figure 8: **Real-world mobile manipulation results.**

References

[1] H. Zhou, R. Wang, Y. Tai, Y. Deng, G. Liu, and K. Jia. You only teach once: Learn one-shot bimanual robotic manipulation from video demonstrations. *arXiv preprint arXiv:2501.14208*, 2025.

[2] H. Kim, J. Kang, H. Kang, M. Cho, S. J. Kim, and Y. Lee. Uniskill: Imitating human videos via cross-embodiment skill representations. *arXiv preprint arXiv:2505.08787*, 2025.

[3] T. G. W. Lum, O. Y. Lee, C. K. Liu, and J. Bohg. Crossing the human-robot embodiment gap with sim-to-real rl using one human demonstration. *arXiv preprint arXiv:2504.12609*, 2025.

[4] P. Dan, K. Kedia, A. Chao, E. W. Duan, M. A. Pace, W.-C. Ma, and S. Choudhury. X-sim: Cross-embodiment learning via real-to-sim-to-real. *arXiv preprint arXiv:2505.07096*, 2025.

[5] C. Wang, L. Fan, J. Sun, R. Zhang, L. Fei-Fei, D. Xu, Y. Zhu, and A. Anandkumar. Mimicplay: Long-horizon imitation learning by watching human play. In *7th Annual Conference on Robot Learning*, 2023.

[6] R.-Z. Qiu, S. Yang, X. Cheng, C. Chawla, J. Li, T. He, G. Yan, D. J. Yoon, R. Hoque, L. Paulsen, et al. Humanoid policy˜ human policy. *arXiv preprint arXiv:2503.13441*, 2025.

[7] Y. Qin, W. Yang, B. Huang, K. Van Wyk, H. Su, X. Wang, Y.-W. Chao, and D. Fox. Anyteleop: A general vision-based dexterous robot arm-hand teleoperation system. *arXiv preprint arXiv:2307.04577*, 2023.

[8] R. Yang, Q. Yu, Y. Wu, R. Yan, B. Li, A.-C. Cheng, X. Zou, Y. Fang, H. Yin, S. Liu, et al. Egovla: Learning vision-language-action models from egocentric human videos. *arXiv preprint arXiv:2507.12440*, 2025.

[9] K. Shaw, S. Bahl, A. Sivakumar, A. Kannan, and D. Pathak. Learning dexterity from human hand motion in internet videos. *The International Journal of Robotics Research*, 43(4):513–532, 2024.

[10] K. Shaw, S. Bahl, and D. Pathak. Videodex: Learning dexterity from internet videos. In *Conference on Robot Learning*, pages 654–665. PMLR, 2023.

[11] Y. Chen, C. Wang, Y. Yang, and K. Liu. Object-centric dexterous manipulation from human motion data. In *8th Annual Conference on Robot Learning*. PMLR, 2024.

[12] K. Li, P. Li, T. Liu, Y. Li, and S. Huang. Maniptrans: Efficient dexterous bimanual manipulation transfer via residual learning. *arXiv preprint arXiv:2503.21860*, 2025.

[13] Z. Mandi, Y. Hou, D. Fox, Y. Narang, A. Mandlekar, and S. Song. Dexmachina: Functional retargeting for bimanual dexterous manipulation. *arXiv preprint arXiv:2505.24853*, 2025.

[14] D. Shah, A. Sridhar, N. Dashora, K. Stachowicz, K. Black, N. Hirose, and S. Levine. Vint: A foundation model for visual navigation. In *Conference on Robot Learning*, pages 711–733. PMLR, 2023.

[15] X. Zhan, L. Yang, Y. Zhao, K. Mao, H. Xu, Z. Lin, K. Li, and C. Lu. Oakink2: A dataset of bimanual hands-object manipulation in complex task completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 445–456, 2024.

[16] R. A. Potamias, J. Zhang, J. Deng, and S. Zafeiriou. Wilor: End-to-end 3d hand localization and reconstruction in-the-wild, 2024.

[17] S. Li, C. Xu, and M. Xie. A robust o (n) solution to the perspective-n-point problem. *IEEE transactions on pattern analysis and machine intelligence*, 34(7):1444–1450, 2012.

[18] B. Wen, W. Yang, J. Kautz, and S. Birchfield. Foundationpose: Unified 6d pose estimation and tracking of novel objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17868–17879, 2024.

[19] AR Code. Ar code, 2022. URL `https://ar-code.com/`. Accessed: 2024-09-28.

[20] A. Handa, K. Van Wyk, W. Yang, J. Liang, Y.-W. Chao, Q. Wan, S. Birchfield, N. Ratliff, and D. Fox. Dexpilot: Vision-based teleoperation of dexterous robotic hand-arm system. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9164–9170. IEEE, 2020.

[21] G. Xu, R. Zheng, Y. Liang, X. Wang, Z. Yuan, T. Ji, Y. Luo, X. Liu, J. Yuan, P. Hua, et al. Drm: Mastering visual reinforcement learning through dormant ratio minimization. In *The Twelfth International Conference on Learning Representations*, 2024.

[22] G. Sokar, R. Agarwal, P. S. Castro, and U. Evci. The dormant neuron phenomenon in deep reinforcement learning. In *International Conference on Machine Learning*, pages 32145–32168. PMLR, 2023.

[23] D. Shah, A. Sridhar, A. Bhorkar, N. Hirose, and S. Levine. Gnm: A general navigation model to drive any robot. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 7226–7233. IEEE, 2023.

[24] A. Sridhar, D. Shah, C. Glossop, and S. Levine. Nomad: Goal masked diffusion policies for navigation and exploration. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 63–70. IEEE, 2024.

[25] Y. Wang, X. He, S. Peng, D. Tan, and X. Zhou. Efficient loftr: Semi-dense local feature matching with sparse-like speed. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 21666–21675, 2024.

[26] M. Zuliani. Ransac for dummies. *Vision Research Lab, University of California, Santa Barbara*, 1, 2009.

[27] K. Madsen, H. B. Nielsen, and O. Tingleff. Methods for non-linear least squares problems. *Informatics and Mathematical Modelling Technical University of Denmark*, 1, 2004.

[28] E. Eade. Gauss-newton/levenberg-marquardt optimization. *Tech. Rep.*, 2013.

[29] M. J. Willis. Proportional-integral-derivative control. *Dept. of Chemical and Process Engineering University of Newcastle*, 6, 1999.

Appendix

The visualization of real-world and simulation results are provided in Figure 9 and Figure 10.

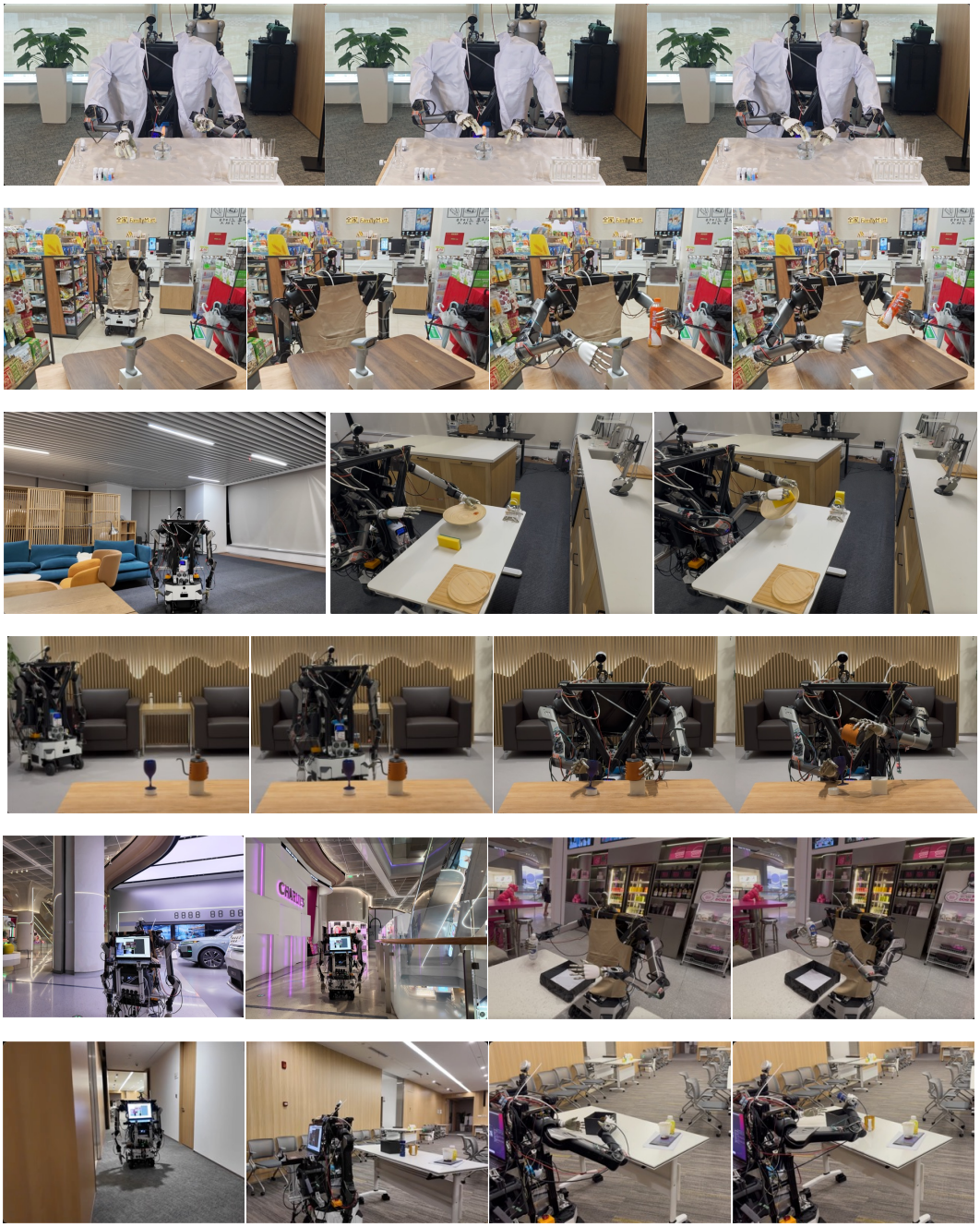

Figure 9: **HERMES exhibits a rich spectrum of mobile bimanual dexterous manipulation skills.** The robot is able to navigate over extended distances in both indoor and outdoor environments, and effectively execute a variety of complex manipulation tasks in unstructured, real-world scenarios, drawing upon behaviors learned from only one-shot human motion.

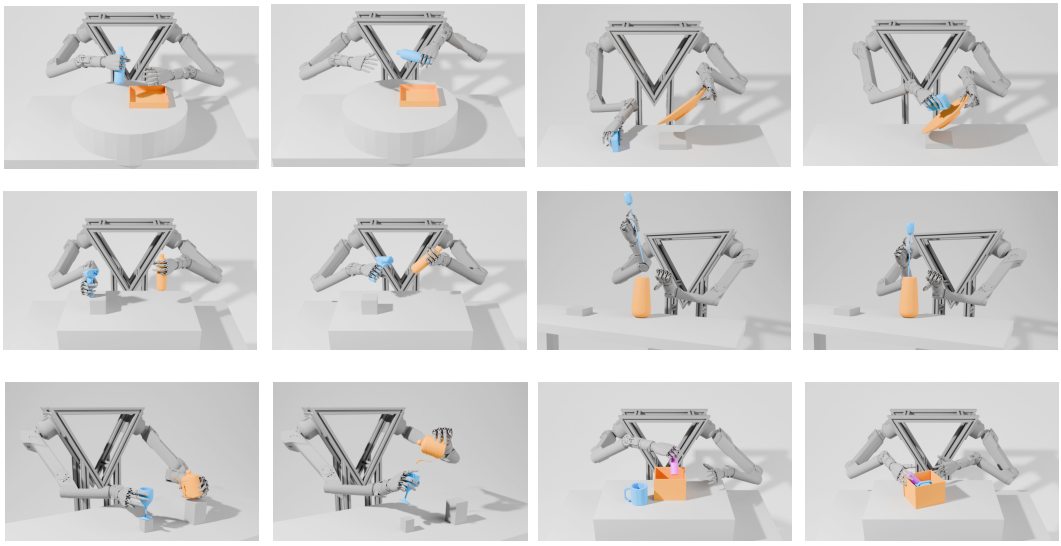

Figure 10: **Simulation training visualization.** We visualize the majority of the training tasks. Leveraging a single reference trajectory in conjunction with a general reward design, HERMES can convert diverse human motion sources into robot feasible behaviors via RL training.