Scale Your Dataset Without Robot

Jiwon Kim* **Kyungzun Rim***
Ilkwon Hong Suhyun Yoon
Robotics LAB, Hyundai Motor Company, Uiwang, South Korea
{robotisme, kyungzun.rim, ikhong, suhyunyoon}@hyundai.com
Authors listed in alphabetical order

Abstract: Imitation learning for robotic manipulation requires extensive demonstration data, yet traditional teleoperation methods are time-consuming, physically constrained, and produce biased datasets. We present a novel VR-based data collection pipeline that addresses these limitations by capturing natural hand demonstrations without robot control. Our approach transforms VR-tracked hand poses into robot-executable trajectories through automated post-processing. We evaluated our method on a real-world task using a Franka Panda manipulator. While the teleoperation-only dataset achieved only 20% success rate due to limited coverage and small dataset size, augmenting it with hand-collected episodes resulted in the combined dataset achieving 63% success rate-a threefold improvement. Notably, our merged dataset matches dual-camera policy performance using only single-image input. Our results demonstrate that VR-based hand demonstrations provide an accessible, efficient solution for scaling robot learning datasets while improving policy generalization and task performance.

Keywords: Imitation Learning, Virtual Reality, Hand Demonstrations

1 Introduction

Generalizability and accuracy are critical criteria for evaluating the performance of imitation learning policies. The number of episodes in datasets is crucial for improving these aspects. However, data collection via teleoperation, which is the most common approach, can be time-consuming and resource-intensive, often requiring specialized infrastructure.

Existing methods face inherent trade-offs: teleoperation provides high-quality data but suffers from time inefficiency and workspace constraints, while non-interactive approaches improve efficiency at the cost of system complexity. Several approaches have attempted to address these challenges. For instance, the ALOHA system [1] uses additional hardware for data collection, but it faces issues with generalizability across different robot types and requires additional physical space. While integrating human demonstrations with rendering or simulations has shown promise, these methods pose difficulties in synchronizing sensor data and often involve complex post-processing to ensure effective policy training.

In this work, we address these challenges by integrating data collection from human demonstrations with teleoperation datasets. Our VR-based method provides a time-efficient way to expand datasets through natural human interactions, enabling unconstrained data collection that seamlessly integrates with standard teleoperation datasets. The main contributions of this paper are:

- **Bridging Gaps Between Human Demonstrations and Robot Movements:** We introduce a transformation pipeline that converts hand poses into gripper poses relative to the robot's base frame. This enables our approach to be applied across diverse robot types, from manipulators to humanoids.

*Equal contribution.

- **Complementary Learning by Leveraging Cross-domain Data:** Human demonstrations and teleoperation datasets differ significantly in aspects such as motion speed, object placement, and camera viewpoint. By merging them into a unified dataset, we obtain policies with substantially improved performance. Notably, the weaknesses of each policy are compensated for when the datasets are combined.
- **Time-efficient and Space-free:** Our method shortens the episode length for data collection by more than 50%. Since only a VR device is required, no additional space or specialized hardware is needed.

2 Related Work

Efficient data collection is crucial for developing generalizable and accurate imitation learning [2] policies in robotics. The quality and diversity of training data directly impact a policy's ability to handle real-world scenarios. This section reviews existing data collection methods, highlighting their limitations and motivating our proposed approach.

2.1 Teleoperated Robot Data Collection

Robot teleoperation has emerged as a standard approach for collecting demonstration data in imitation learning. Direct manipulation using VR controllers or 3D space mice provides intuitive control and naturally generates robot-executable trajectories. However, teleoperation suffers from significant time inefficiency, as collecting sufficient demonstration data requires extensive human effort. Moreover, dataset quality varies considerably based on operator expertise and fatigue. Several approaches have attempted to improve teleoperation efficiency through specialized hardware systems. ALOHA[1] addresses this challenge through a leader-follower setup where operators manipulate a kinematically identical robot structure, providing natural haptic feedback and real-time response that accelerates data collection. Alternative approaches leverage hand tracking for robot control through VR devices [3, 4, 5] or vision-based systems [6]. While such methods are cost-effective and applicable to different types of robots, they suffer from inaccuracies in hand-tracking and mismatches in action ranges between human and robot, both of which degrade the quality of the collected data and, consequently, the trained policy. More critically, the coupling of data collection with real-time robot control constrains operators to the robot's reachable workspace and movement dynamics, limiting both the spatial diversity and naturalness of demonstrations.

2.2 Non-Interactive Data Collection

Non-interactive methods collect human demonstration data without real-time robot control, offering potential efficiency gains. The primary challenge in these approaches is mapping between human hand movements and robot-executable actions.

Chi et al. [7] address this challenge using visual SLAM to align hand trajectories, captured while manipulating gripper-shaped tools, with corresponding robot poses. However, their approach requires complex multi-sensor synchronization and extensive pre- and post-processing pipelines. Other human-data-driven approaches employ alternative strategies, such as hierarchical learning with coarse-to-fine refinement or reinforcement learning augmented with large-scale synthetic data [8, 9]. These methods, while effective, require either complex multi-stage training pipelines or extensive computational resources for synthetic data generation.

In contrast, our VR-based approach enables efficient dataset expansion with minimal post-processing. By using the VR device as a single, unified data source, we eliminate synchronization issues while maintaining compatibility with standard imitation learning pipelines. The collected hand demonstrations can be directly combined with teleoperation data, enhancing policy performance without requiring specialized training procedures.

3 Implementation

This section describes the implementation of our VR-based data collection system and the post-processing pipeline that transforms hand demonstrations into robot-executable trajectories. We first detail the data collection process, then explain the post-processing steps that generate robot-compatible datasets, and finally describe the policy training approach.

3.1 Collection of Hand Pose Data

3.1.1 Data Collection Process

The data collection pipeline begins with a one-time calibration to establish the robot base frame in the VR coordinate system, essential for the transformations described in Section 3.2.1. Since the exact position and orientation of the robot base frame cannot be directly accessed from outside the hardware, we instead rely on surrounding points that satisfy two conditions: (1) their exact pose can be obtained in the VR frame, and (2) their geometric relationship to the origin of the robot base frame is known. We select the four mounting holes surrounding the robot base as reference points. The operator marks the 6-DoF poses of these mounting points using VR controllers. With this information, ${}^{V}H^{B}$, the pose of the robot origin with respect to the VR origin is calculated based on the known relationships provided in the robot datasheet [10].

Once the base frame is established, we proceed to episode collection. Pinch gestures are used to signal the start and end of an episode, eliminating the need for additional input devices and making the process more convenient for the human expert.

3.1.2 Hand Pose Definition

Converting hand-tracking data to robot commands requires defining a mapping from the 26 tracked hand bones to a single end-effector pose. We construct a hand frame H that corresponds to the robot gripper configuration.

The origin of H is set at the wrist, as it is tracked with high confidence and naturally serves the same role as the flange of a manipulator. Therefore, the translational component of H is taken to be the same as that of the wrist bone. For the rotational component, we assume that the hand is posed to resemble a typical parallel-jaw gripper—that is, the thumb and index finger point in the same direction while the remaining fingers are slightly curved inward, as described in Figure 1. This assumption aligns with natural human grasping postures for manipulation tasks. Under this assumption, align the axes of H to match the gripper coordinate system. Specifically, since the z-axis of the gripper we use (Franka Hand) points toward the grasp target, the z-axis of H points from the wrist toward the midpoint between thumb and index fingertips, which can be regarded as the target of the hand in a grasping task. The remaining axes are defined in a similar manner, following the right-hand rule.

Importantly, this definition of hand pose is not fixed. It can be redefined to match the kinematics of any robot or gripper. This flexibility means that the collected frames are not tied to a single embodiment but can instead serve as a generalizable source of training data applicable across a wide variety of robot platforms.

3.2 Post-Processing of Data

Datasets collected in real time undergo post-processing to construct robot-ready datasets for imitation learning. The key discrepancies to be addressed are the mismatch in coordinate frame origins and the absence of certain control-relevant information in human hand data.

3.2.1 Frame Transformation

The recorded trajectories contain the homogeneous transformation from the VR frame origin to the human hand at each timestep. However, to train and deploy a policy, the end-effector pose must

Figure 1: **Example definition of a hand frame.** (Left) Frames in gray represent raw hand-tracking outputs, i.e., the 6-DoF poses of each bone in a human hand. The RGB-colored frame shows the defined hand frame, using the Franka Hand frame (Right) as a reference.

Figure 2: **Hand images corresponding to different gripper states.** For evaluation of current gripper states, images were fed into VLM with prompt "Analyze this image and determine the state of the hand's grasp on the bottle. Answer with a float from 0.0 (fully closed) to 1.0 (fully open).". Scores are 1.0, 0.7, and 0.1 respectively for each image.

be expressed in the robot base frame. Therefore, the saved trajectories need to be converted from the VR frame to the robot base frame. The pose of the end-effector ${}^{B}H_t{}'$ at time t is calculated using Equation 1, where ${}^{V}H_t$ is the homogeneous transform to the hand at time t, and ${}^{V}H^{B}$ is the homogeneous transform to the robot base frame B, both from the VR frame V.

$$ {}^{B}H_t{}' = {}^{V}H^{B^{-1}} \times^{V} H_t \tag{1} $$

Note that ${}^{V}H_t$ represents the wrist pose, so ${}^{B}H_t{}'$ corresponds to the robot's last link rather than the gripper. In contrast, the trajectories recorded in existing teleoperation datasets are gripper poses, with the gripper attached to the last link by a fixed translational and rotational offset. Therefore, for consistency, the transformation from the last link of the robot to the gripper—whose precise values are provided by the gripper's manufacturer [11]—is applied to obtain the final trajectory ${}^{B}H_{t=1,\ldots,T}$.

3.2.2 Gripper Value Generation

In order to train a policy for a robot with parallel-jaw gripper, 1 DoF action command is necessary to complete the task. Since hand demonstrations lack explicit gripper commands, we employ Vision Language Models (VLMs) [12, 13] to classify hand states as open or closed grips. When the state changes from closed to open (or vice versa), it is recorded as a discrete action.

To improve robustness against VLM prediction errors, we adopt two strategies. First, the moving average over N frames was used to decide the state transition. Also, under the assumption that holding or releasing of an object can only happen when the hand movement is sufficiently slow, the linear velocity of hand is used as an additional filter to confirm state changes. As a result, we achieved 95 % accuracy in estimating gripper states from the images using VLMs. Sample VLM inferences and the whole process is illustrated in 2 and C, respectively.

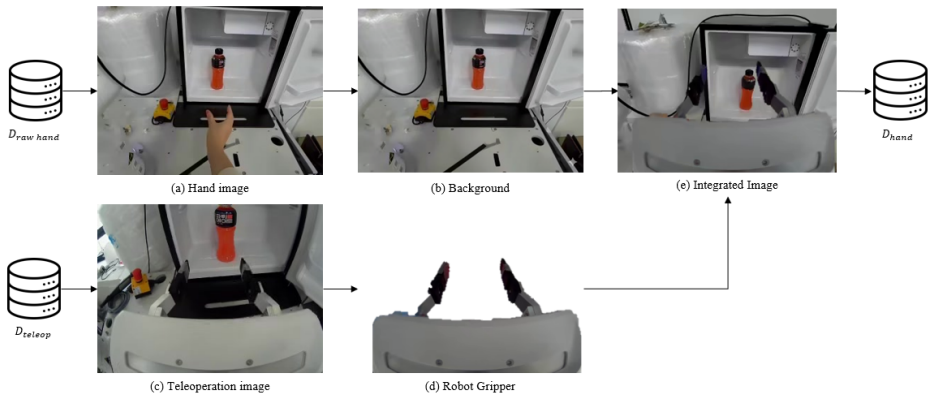


Figure 3: **Visual Gap Adaptation** To bridge the visual gap between human demonstrations and robot deployment, we modify collected images through a three-stage process. (a) Original hand demonstration image. (b) Background image after human hand removal via generative inpainting. (c) Reference teleoperation image containing the robot gripper. (d) Segmented gripper mask extracted from the teleoperation dataset. (e) Final composite image combining the inpainted background with the robot gripper, used for policy training.

3.2.3 Image Modification

A critical domain gap between hand demonstrations and robot deployment arises from visual observations: while robot-mounted cameras capture the robotic gripper, hand demonstration datasets contain human hands. This visual discrepancy can significantly impact policy performance during deployment.

To address this gap, we leverage existing teleoperation datasets collected on real robots. Since wrist-mounted cameras maintain a consistent view of the robot gripper across teleoperation episodes, we extract and integrate these gripper views into hand demonstration images. Our approach consists of three stages: (1) segmentation of the robot gripper from teleoperation images, (2) removal of the human hand from demonstration images using generative inpainting, and (3) composition of the extracted gripper onto the inpainted background at the appropriate position. This process, illustrated in Figure 3, produces training images that closely resemble those encountered during robot deployment while preserving the original scene context and task-relevant objects.

3.2.4 Other Modifications

We applied several additional modifications to improve the quality and consistency of the training data.

The recorded hand trajectories include subtle, unintended movements immediately after the starting pinch and before the ending pinch. These hesitation movements can degrade the quality of the training data and subsequently affect the learned policy. To address this, we applied two trimming strategies. First, in the beginning of each episode, a predefined number of frames was discarded, corresponding to the average initialization time required for the hand trajectory after the starting pinch. Next, at the end of each episode, we took a more adaptive approach of removing all frames after the task completion, which is also judged by a VLM.

Second, to ensure consistency across the dataset, we applied additional filtering to eliminate episodes that significantly differ from rest. Since the main causes of such differences were the misinterpreted gestures and incorrect judgements by the VLM, 3 frames were investigated on each episode: the start, the end, and the grip-the moment when the expert grabs the target object, corresponding to closing gripper action of a robot. This process identified 5% of episodes with incorrect grip judgments and 2% with gesture recognition errors, resulting in 7% of collected data being excluded from training. Lastly, in our dataset, we also include joint trajectories, which often shows promising re-

sults in the trained policy. Without actual robot execution, we compute joint configurations through that minimize trajectory smoothness while avoiding singularities through null-space optimization. The resulting trajectory was verified on simulation and real robot to make sure that every joint state is reachable. Additional visualization results are provided in Appendix B.

3.3 Training a Policy

In our work, the teleoperation dataset contains only 46 episodes, which is insufficient for training large VLA models such as [14] [15]. We evaluated two models: SmolVLA[16] and our custom VLA model, a sample-efficient dual-system model inspired by [17]. Due to SmolVLA's poor performance on our limited dataset, we focus our evaluation on the custom VLA to assess the impact of hand-collected data on policy performance.

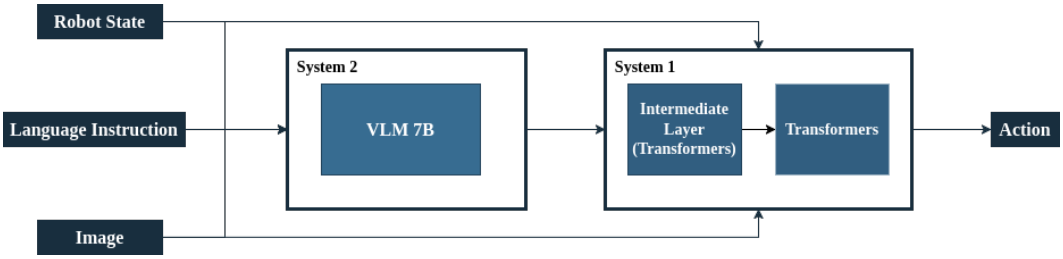


Figure 4: Custom VLA Architecture

4 Experimental Results

4.1 Experimental Setup

Images and hand poses were transmitted via Wi-Fi from a custom VR application at 15 Hz. On the PC side, frames were sampled at 10 Hz, each consisting of a single image from the front-facing camera of the Meta Quest 3 and the 6-DoF pose of the human hand in the VR coordinate frame. The trained policy was deployed on a Franka Panda, a 7-DoF manipulator mounted on a tabletop setup, with inference performed on an NVIDIA A6000 GPU.

4.2 Datasets

Our work utilizes three types of datasets: one collected via teleoperation, another using hand poses, and a third that integrates both. Sample images from each dataset are shown in Appendix A. To evaluate the effect of the additional hand-pose data, we first defined six sections on the refrigerator's surface, as shown in Figure 5a. By counting the number of episodes where target objects fall within each section, we visualized the distribution of target object poses along with the summary of datasets in Figure 5.

The teleoperation dataset consists of 46 episodes. However, none of these episodes feature a target object located in Section 1 or Section 4. To compensate for this imbalance, the hand-pose dataset includes many episodes with target objects specifically placed in these two sections, making the datasets complementary to each other.

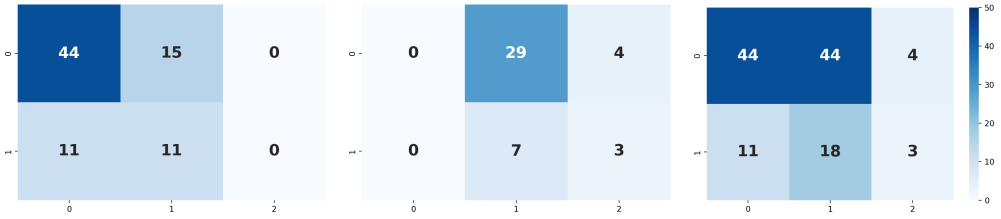
4.3 Policy Performance

We evaluated policy performance on the "Take bottle from fridge" task using various training datasets. Table 1 summarizes the success rates for each refrigerator section across different data configurations. The policy trained on the expanded dataset (P_{T+H}), which combines teleoperation and hand-motion data, achieved the highest overall success rate (63%)—a threefold improvement over the teleoperation-only baseline (P_T). This dramatic improvement demonstrates how the hand

(a) Definition of Target Object Section

Dataset	# Episodes	Avg. Episode Length (s)
Teleoperation	46	17.13
Hand	83	5.88
Merged	129	9.89

(b) Summary of Datasets



(c) Hand (d) Teleoperation (e) Merged

Figure 5: **Dataset characteristics and target object distribution.** (a) Definition of six sections within the 18 cm × 37 cm refrigerator area used for analysis. (b) Summary statistics showing the number of episodes and average completion time for each dataset, highlighting the faster execution in hand demonstrations. (c-e) **Heatmaps showing the spatial distribution of target object placements** across datasets. Each cell value represents the number of episodes with the target object in that section. The hand dataset provides better coverage in sections 1 and 4, while teleoperation data concentrates in sections 2, 3, and 5. The merged dataset achieves more uniform coverage across all sections. Note that 2 hand episodes and 3 teleoperation episodes with objects outside the defined sections are excluded from these visualizations.

dataset effectively compensates for the limitations of the existing teleoperation data. We identify two key complementary factors:

1. Expanded spatial coverage: The teleoperation dataset had limited coverage in certain regions due to the constraints of robot operation. The hand dataset, collected without robot constraints, naturally covered these underrepresented areas. For sections 1 and 4—regions poorly represented in teleoperation data—P_{T+H} shows substantial gains, demonstrating that hand data successfully fills the gaps in spatial distribution.

2. Enhanced motion dynamics: The hand dataset provides larger, more decisive movements that are difficult to achieve through teleoperation. As shown in Table 5b, hand demonstrations complete tasks faster with more aggressive motions. This directly addresses the critical failure mode of P_T, where teleoperation's cautious, small movements often failed to reach precise grasping poses when near the target. The hand dataset's natural, unconstrained motions effectively complement this weakness.

Notably, this merged approach achieves performance equivalent to policies trained with dual-camera inputs, despite using only single-image input. This demonstrates that our data collection approach effectively compensates for the limitations of single-view teleoperation data, providing a more practical, robust and cost-effective solution than adding additional sensors.

5 Conclusion

In this paper, we presented a data collection pipeline that makes scaling datasets both easy and intuitive. When combined with even a small amount of teleoperation data, it significantly enhances policy capabilities. Specifically, our experiments demonstrate that the task completion rate increased approximately threefold compared to the baseline, achieving performance comparable to policies with additional sensor inputs. While the proposed method shows promising results, we acknowledge several limitations inherent to hand data collection.

Table 1: Success Rate of Policies Trained on Different Datasets

	1 Image Input			2 Image Input
	Teleoperation	**Hand**	**Teleoperation + Hand (expanded dataset)**	**Teleoperation**
Section 1(5)	0/5(0%)	0/5(0%)	**3/5(60%)**	2/5(40%)
Section 2(5)	0/5(0%)	0/5(0%)	**4/5(80%)**	5/5(100%)
Section 3(5)	0/5(0%)	0/5(0%)	**2/5(40%)**	3/5(60%)
Section 4(5)	0/5(0%)	0/5(0%)	**3/5(60%)**	0/5(0%)
Section 5(5)	3/5(60%)	0/5(0%)	**4/5(80%)**	4/5(80%)
Section 6(5)	3/5(60%)	0/5(0%)	**3/5(60%)**	5/5(100%)
Success Rate	6/30(20%)	0/30 (0 %)	**19/30(63%)**	19/30(63%)

Note: A policy trained with an additional image input (third-eye view) provided as a reference.

The applicability of this approach is constrained by the inherent challenges of VR device hand tracking. Users must be cautious in scenarios where hands are frequently occluded or under low-light conditions—both well-known limitations of most hand-tracking systems. Future work could explore generating smooth trajectories from a limited number of high-confidence hand poses.

While we addressed the primary visual domain gap through gripper replacement, our current approach relies on fixed gripper views extracted from teleoperation data, which does not account for the varying hand poses and gripper orientations during demonstrations. Additionally, perspective differences between the VR device's front-facing camera and the robot's wrist-mounted fisheye camera remain unaddressed. A more sophisticated approach would involve dynamically rendering a 3D gripper mesh based on real-time hand pose, creating more realistic visual correspondences between training and deployment.

Finally, our approach to defining the rotational component of hand poses has limitations. As we adopted a relatively naive method for generating poses from raw hand-tracking data, future work will explore using other skeletal features with higher tracking accuracy than fingertips.

Acknowledgments

This work is supported by Robotics Vision AI team of the Robotics LAB. We thank Dr. Dongjin Hyun, the leader of the Robotics LAB and Dr. Jaeho Lee, the leader of the Robotics Vision AI team. Also, we also acknowledge support of our colleagues including Dongheon Shin and Jaekwang Cha.

References

[1] A. . Team, J. Aldaco, T. Armstrong, R. Baruch, J. Bingham, S. Chan, K. Draper, D. Dwibedi, C. Finn, P. Florence, S. Goodrich, W. Gramlich, T. Hage, A. Herzog, J. Hoech, T. Nguyen, I. Storz, B. Tabanpour, L. Takayama, J. Tompson, A. Wahid, T. Wahrburg, S. Xu, S. Yaroshenko, K. Zakka, and T. Z. Zhao. Aloha 2: An enhanced low-cost hardware for bimanual teleoperation, 2024. URL https://arxiv.org/abs/2405.02292.

[2] A. Hussein, M. M. Gaber, E. Elyan, and C. Jayne. Imitation learning: A survey of learning methods. *ACM Computing Surveys (CSUR)*, 50(2):1–35, 2017.

[3] O. Kwon, S. Yamsani, N. Myers, S. Taylor, J. Hong, K. Park, A. Alspach, and J. Kim. Paprle (plug-and-play robotic limb environment): A modular ecosystem for robotic limbs, 2025. URL https://arxiv.org/abs/2507.05555.

[4] A. Iyer, Z. Peng, Y. Dai, I. Guzey, S. Haldar, S. Chintala, and L. Pinto. Open teach: A versatile teleoperation system for robotic manipulation, 2024. URL https://arxiv.org/abs/2403.07870.

[5] R. Ding, Y. Qin, J. Zhu, C. Jia, S. Yang, R. Yang, X. Qi, and X. Wang. Bunny-visionpro: Real-time bimanual dexterous teleoperation for imitation learning, 2024. URL https://arxiv.org/abs/2407.03162.

[6] 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, 2020. doi:10.1109/ICRA40945.2020.9197124.

[7] C. Chi, Z. Xu, C. Pan, E. Cousineau, B. Burchfiel, S. Feng, R. Tedrake, and S. Song. Universal manipulation interface: In-the-wild robot teaching without in-the-wild robots, 2024. URL https://arxiv.org/abs/2402.10329.

[8] J. Yu, L. Fu, H. Huang, K. El-Refai, R. A. Ambrus, R. Cheng, M. Z. Irshad, and K. Goldberg. Real2render2real: Scaling robot data without dynamics simulation or robot hardware, 2025. URL https://arxiv.org/abs/2505.09601.

[9] Y. Chen, C. Wang, Y. Yang, and K. Liu. Object-centric dexterous manipulation from human motion data. In *8th Annual Conference on Robot Learning*, 2024. URL https://openreview.net/forum?id=KAzku0Uyh1.

[10] F. E. GmbH. *Franka Emika Robot's Instruction Handbook*, 2021. URL https://download.franka.de/documents/100010_Product%20Manual%20Franka%20Emika%20Robot_10.21_EN.pdf.

[11] F. E. GmbH. *Franka Hand Product Manual*, 2022. URL https://download.franka.de/documents/220010_Product%20Manual_Franka%20Hand_1.2_EN.pdf.

[12] OpenAI. Gpt-4 technical report, 2024. URL https://arxiv.org/abs/2303.08774.

[13] G. Team. Gemini: A family of highly capable multimodal models, 2025. URL https://arxiv.org/abs/2312.11805.

[14] K. Black, N. Brown, D. Driess, A. Esmail, M. Equi, C. Finn, N. Fusai, L. Groom, K. Hausman, B. Ichter, et al. $\pi 0$: A vision-language-action flow model for general robot control. corr, abs/2410.24164, 2024. doi: 10.48550. *arXiv preprint ARXIV.2410.24164*, 2024.

[15] M. J. Kim, K. Pertsch, S. Karamcheti, T. Xiao, A. Balakrishna, S. Nair, R. Rafailov, E. Foster, G. Lam, P. Sanketi, et al. Openvla: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024.

[16] M. Shukor, D. Aubakirova, F. Capuano, P. Kooijmans, S. Palma, A. Zouitine, M. Aractingi, C. Pascal, M. Russi, A. Marafioti, S. Alibert, M. Cord, T. Wolf, and R. Cadene. Smolvla: A vision-language-action model for affordable and efficient robotics, 2025. URL https://arxiv.org/abs/2506.01844.

[17] Figure AI. Helix: A vision-language-action model for generalist humanoid control, 2025. URL https://www.figure.ai/news/helix.

Appendix

A Images in Datasets

We provide snapshots of video saved in a sample episode of each dataset.

A.1 Hand

Figure A.1: Human hand demonstration frames (t=1.0s to t=5.0s at 1.0s intervals).

A.2 Teleoperation

These images are from the only camera view that our baseline teleoperation dataset contains.

Figure A.2: Teleoperation demonstration frames from robot's wrist camera (t=3.0s to t=15.0s at 3.0s intervals).

A.3 Additional Input Image for Teleoperation

The dual-camera dataset referenced in Table 1 combines views from both Figures A.2 and A.3.

Figure A.3: Teleoperation demonstration frames from third-person view camera (t=3.0s to t=15.0s at 3.0s intervals).

B Verification of collected trajectories

After post-processing the collected hand data, we verified the resulting trajectories in both simulation and on a real robot. The figures below show the same episode from Section A.1 replayed in different environments.

B.1 Simulation

We utilized PyBullet for simulation with a URDF model of the Franka Panda robot.

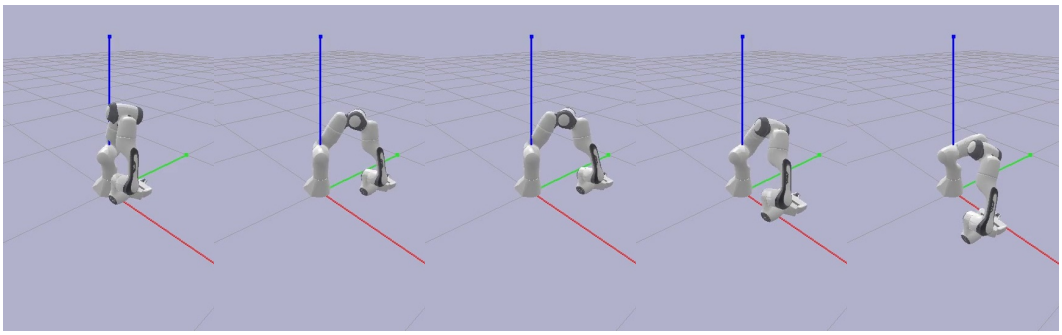


Figure A.4: Replay of hand demonstration trajectory in PyBullet simulation with Franka Panda robot (t=1.0s to t=5.0s at 1.0s intervals).

B.2 Real Robot

Absolute joint positions from the processed trajectories were executed on a Franka Panda robot for validation.

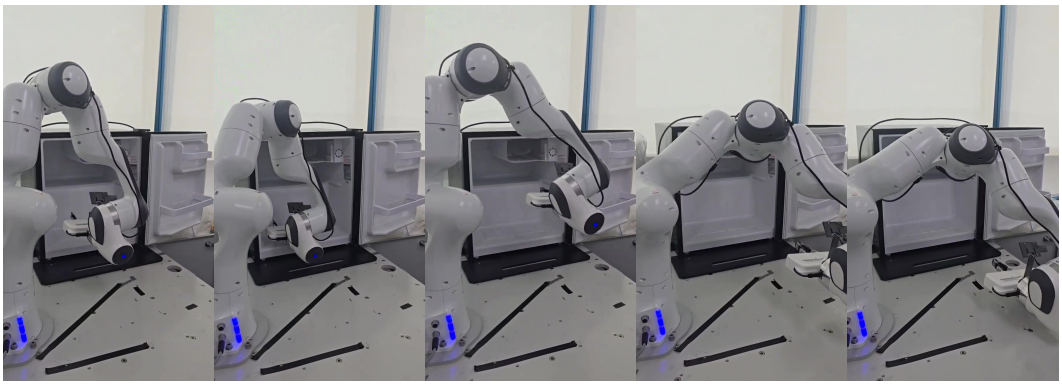


Figure A.5: Replay of hand demonstration trajectory on real Franka Panda robot (t=1.0s to t=5.0s at 1.0s intervals).

C Gripper Value Generation Process

Images saved in the raw hand dataset are used for gripper state judgement through 2-step process. First, a VLM decides if the hand in the image corresponds to closed or open gripper state, then it goes through multi-modal filtering. The final score is used to determine gripper states and actions that will be saved as a part of dataset.

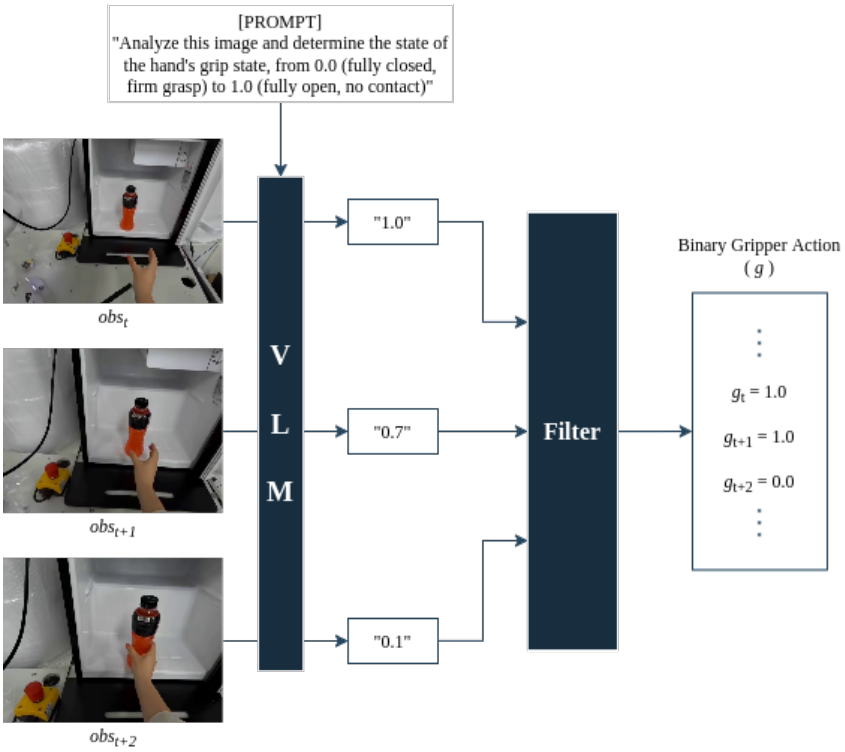


Figure A.6: **Gripper Value Generation**