Toward Motion Diverse Community Datasets for Scaling VLA Pretraining

Anonymous CVPR submission

Paper ID *****

Abstract

Vision-Language-Action (VLA) models have recently benefited from large-scale pretraining strategies inspired by advances in language and vision. However, unlike text and image domains, robotics remains constrained by limited embodied interaction data. While community efforts such as LeRobot community datasets have improved accessibility and standardization, existing datasets are often dominated by structurally similar, mostly pick and place tasks, limiting diversity in motion primitives. In this work, we investigate the role of motion diversity in scaling community collected VLA pretraining data. We introduce a manipulation dataset, called HeteroMotion, comprising 15 tasks across five behavior categories and 1,050 trajectories designed to expand action space coverage and reasoning complexity. Through controlled scaling HeteroMotion improves downstream real-world performance compared to direct fine-tuning or small-scale pretraining. A joint variance analysis further reveals that HeteroMotion, provides broader motion coverage across all robot joints relative to existing community datasets.

1. Introduction

In recent years, there has been growing interest in transferring techniques across natural language processing and computer vision through the adoption of large-scale pretrained models [16, 36]. These models, trained on vast corpora of text [9], images [34], image-text pairs [36], and audio data [20, 43], have demonstrated strong cross-domain transfer capabilities [1, 37]. The availability of large, diverse datasets has enabled scaling laws to emerge, allowing data-intensive algorithms to achieve remarkable performance improvements [16, 22].

In contrast, robotics has not benefited from the same abundance of large-scale data. The limited availability of diverse robotic interaction data has created a significant bottleneck for adopting data-intensive approaches that scale well but require substantial training data [12, 35].

This issue has largely arisen due to the time-consuming and costly nature of robotic data collection, which primarily relies on humans teleoperating expensive robots. Although it is possible to train a model with only an action expert [39, 45], recent work typically uses a pretrained Vision-Language Model (VLM) as the backbone of VLA systems to leverage large-scale visual and linguistic pretraining [3, 4, 6, 8, 14, 17, 19, 23, 25, 26, 29, 38]. While this approach has led to remarkable improvements, prior work has noted that robotic performance still remains fundamentally constrained by the scale and diversity of embodied interaction data available during training [8, 35]. In other words, while the VLM's perceptual abilities transfer effectively, the ability to generalize actions is still limited by the available robotic training data.

The open-source community has approached the challenge of obtaining large-scale robotic data through two primary strategies. The first approach involves multi-institutional collaborations that collect teleoperated data across different embodiments. Prominent examples include Open X-Embodiment (OXE) [35], with more recent efforts such as DROID [24] and Bridge [15]. These datasets have powered many recent VLA models; however, they exhibit three main limitations. First, they are not highly extensible. After the initial data collection phase, further additions are rare, leading to the formation of "data islands." Second, they are often not collected through a unified framework, resulting in significant data cleaning, episode filtering, and engineering efforts before they can be effectively used. This issue is particularly evident in OXE, where prior works have selected their own subsets [25]. Although newer datasets such as DROID mitigate some of these concerns, standardization remains an important consideration for public datasets. Third, the cost of entry is high: these datasets typically rely on expensive robotic platforms (often exceeding $30,000 [2]), which naturally limits broader community participation.

The second approach focuses on community-driven datasets, with LeRobot [11] being the most prominent example. This approach addresses many of the limitations of the first strategy. It provides a unified and extensible data collection framework, supports integration of new robots,

and remains fully open source. Additionally, it leverages affordable robotic platforms (approximately $300 [10]), significantly lowering the barrier to entry for contributors. Previous work has demonstrated that models trained solely on community-collected data can achieve competitive performance [38].

However, this accessibility introduces its own trade-offs. First, affordable platforms such as SO101 have a more limited range of motion compared to higher-end systems [2]. Second, many community-collected datasets consist primarily of simple, short-horizon pick-and-place tasks. While such tasks are useful for adapting models to new environments and setups, they do not sufficiently address the fundamental issue of motion diversity. As a result, scaling pretraining on structurally similar tasks may yield diminishing performance gains [41].

Our work aims to address this second limitation. We first analytically demonstrate the prevalence of similar tasks within the existing dataset. We then introduce HeteroMotion, a motion-diverse robotic manipulation dataset that expands action-space coverage. It includes heterogeneous motion primitives and interactions beyond simple tasks. Finally, through real-world experiments, we show that pretraining on this dataset leads to measurable improvements compared to models trained without diverse pretraining.

2. Related Work

Recent progress in VLA models [19, 25, 38] has been closely tied to advances in large-scale robotic data collection [7, 8]. Existing approaches primarily differ in how data is gathered, standardized, and shared across institutions and embodiments [24, 35, 40]. While these efforts have significantly improved policy learning, they introduce trade-offs between accessibility, diversity, and extensibility. In this section, we position our work within these data paradigms filling the remaining gaps in action-space diversity.

Data Requirements in VLA Training In the recent development of VLA models, the general training recipe has followed the pretraining/post-training paradigm established in large language and vision models [9, 16, 36]. As a result, two categories of data are required for this type of training. The first is large-scale pretraining data, which usually consists of multiple embodiments performing diverse tasks across different environments [24, 35]. This helps the model generalize, recover from mistakes, and adapt to different embodiments [7]. The second category is high-quality, task-specific fine-tuning data. Fine-tuning data is typically embodiment and task specific; therefore, much of the community effort has focused on creating large pretraining datasets, or incorporating fine-tuning data into larger community datasets.

Limitations of Existing Large-Scale VLA Datasets There are examples of closed-source VLA models where the training data is not public [18], preventing future work from leveraging these datasets as part of their pretraining regimen. However, there are also notable efforts aimed at gathering large-scale pretraining data. Prominent examples include OXE [35], DROID [24] , and Bridge [40], all of which have been repeatedly used in recent work to develop foundation VLA models. Nevertheless, three main challenges remain. First, some of the data was not collected using a unified platform, resulting in inconsistencies in control frequency and data quality. Consequently, substantial data engineering is required to extract high-quality pretraining data. Second, this line of work is not easily extensible, as it primarily relies on academic-industrial collaborations to create large-scale datasets, leading to the formation of data islands [21]. Finally, data collection in these datasets depends on expensive robotic infrastructure, which creates a barrier to entry for broader community participation.

LeRobot Community Dataset LeRobot aims to address two of these main challenges. First, it provides an extensible framework that supports the integration of new datasets and robotic platforms. Second, it lowers the barrier to entry by offering support for affordable robots such as SO100, SO101, and Koch v1.1 [10]. Previous work has demonstrated that pretraining on the available LeRobot dataset yields measurable performance improvements [38]. However, upon closer inspection of the latest version of the dataset, we find that out of 791 total tasks, approximately 500 correspond to variations of pick-and-place behaviors that are highly similar in their underlying motion primitives. Our work explicitly aims to address these issues and propose a path toward higher-quality, community-collected VLA pretraining data with increased motion diversity.

3. Methodology

Our work directly addresses the limitations of motion homogeneity in existing community datasets. We introduce a structured collection of 15 tasks spanning five behavior categories and, more importantly, through a controlled quantitative scaling study, demonstrate that increasing structured motion diversity in community-collected data leads to measurable improvements in real-world downstream performance across diverse manipulation settings.

3.1. Dataset

Figure 1 illustrates HeteroMotion, which consists of 15 tasks grouped into five behavior categories, with three tasks per category and 50 demonstrations per task.

To increase motion diversity and reasoning complexity beyond repetitive short-horizon pick-and-place, we design the following categories:

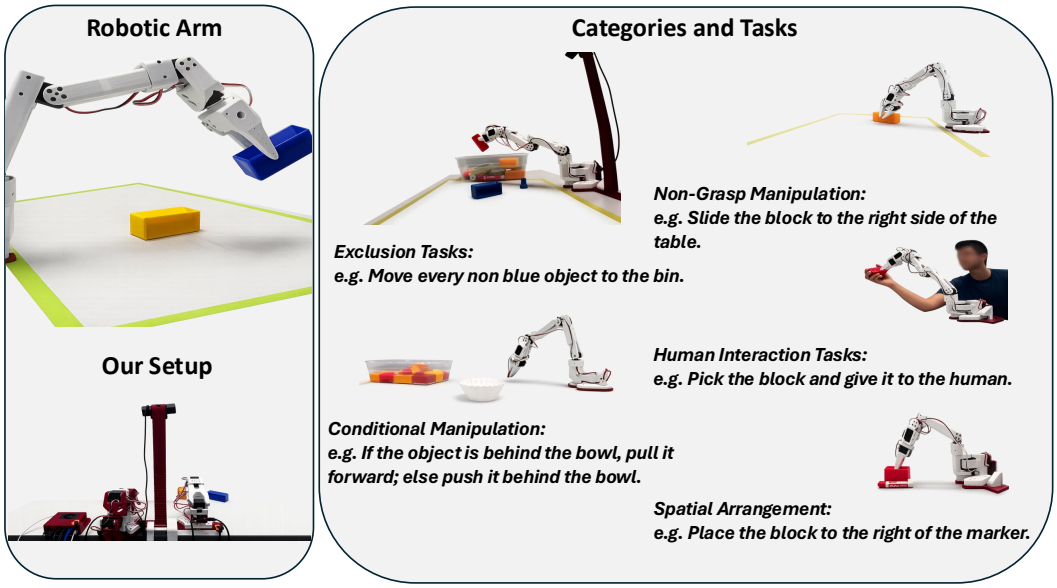


Figure 1. Overview of the proposed dataset, HeteroMotion, highlighting the robotic platform and five task categories designed to promote diverse manipulation skills and complex reasoning.

1. **Non-Grasp Manipulation:** Focuses on actions such as pushing or sliding that do not involve gripper actuation.
2. **Conditional Manipulation:** Tasks that require reasoning over conditional instructions prior to execution.
3. **Human Interaction:** Includes direct human–robot interaction scenarios, such as object handover. The robot must account for human presence and coordinate its motion accordingly.
4. **Spatial Arrangement:** Involves arranging objects according to specified spatial relationships (e.g., left of, behind, inside).
5. **Exclusion Tasks:** This category requires the robot to deliberately ignore specified objects while manipulating others.

A complete breakdown of all 15 tasks and their descriptions is provided in Appendix A (Tab. 2).

These five categories form the primary pretraining dataset. Additionally, we provide two supplementary datasets: (1) 150 autonomously generated, human labeled trajectories to support reinforcement learning methods such as HIL-SERL [31], and (2) 150 simple manipulation trajectories from downstream tasks introduced in experiments. HeteroMotion contains approximately 1,050 manipulation trajectories for VLA pretraining.

3.2. Training

To validate the effectiveness of HeteroMotion in improving downstream VLA performance and to study scaling effects, we trained five model variants fine-tuned on downstream tasks. These include the original SmolVLA [38] model and four additional variants of the same model pretrained on 10%, 25%, 50%, and 100% of HeteroMotion.

We ensured that the smaller pretraining subsets were sampled uniformly and maintained task diversity. Each variant was then fine-tuned for three downstream tasks: (1) pick-and-place, (2) sorting, and (3) stacking. All models were trained for a proportional number of steps, corresponding to an equal number of epochs, followed by 20k steps for task adaptation. We used the LeRobot training framework [11], initializing from *lerobot/smolvla_base*. During training, only the action expert was trained while the backbone VLM remained frozen.

The action expert models the distribution $p(\mathbf{A}_t|\mathbf{o}_t)$ where $\mathbf{A_t}$ corresponds to an action chunk [45] for H actions and hence $\mathbf{A}_t = [\mathbf{a}_t, \mathbf{a}_{t+1}, \ldots, \mathbf{a}_{t+H-1}]$. Our observation on the other hand is $\mathbf{o_t} = [\mathbf{I_1}, \mathbf{I_2}, l_t, \mathbf{q_t}]$ where l_t correspond to the language prompt, $\mathbf{q_t}$ to the robot's joints and $\mathbf{I_1}$ and $\mathbf{I_2}$ correspond to top and side camera respectively. We used the following flow matching loss, introduced in [27, 30] and later adopted in [5]:

$$L^\tau(\theta) = \mathbb{E}_{p(\mathbf{A_t}|\mathbf{o_t}),q(\mathbf{A_t^\tau}|\mathbf{A_t})}||\mathbf{v}_\theta(\mathbf{A}_t^\tau, \mathbf{o}_t) - \mathbf{u}(\mathbf{A}_t^\tau|\mathbf{A}_t)||^2$$

where the subscripts correspond to trajectory timesteps (t) and the superscripts correspond to the flow-matching timesteps (τ). The network works by sampling an $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, computing the noisy actions $\mathbf{A}_t^\tau = \tau\mathbf{A}_t + (1-\tau)\epsilon$, and then training $\mathbf{v}_\theta(\mathbf{A_t^\tau}, \mathbf{o}_t)$ to match the denoising vector field $\mathbf{u}(\mathbf{A}_t^\tau|\mathbf{A}_t) = \mathbf{A}_t - \epsilon$.

All experiments were conducted on a single A100 GPU. Training required approximately one hour per 5k steps, to-

taling roughly 100 GPU hours for the project.

4. Evaluation and Results

All experiments were performed on the SO101 robotic arm platform. The system utilized two RGB cameras: a top-down camera for global scene observation and a side-view camera to capture interaction dynamics. Both cameras operated at a resolution of 640 × 480 pixels. This dual-view configuration provides complementary spatial perspectives, enhancing manipulation robustness and mitigating errors caused by occlusions. Experiments were conducted on a Jetson Orin Nano (8GB) [33], an Ampere-based edge AI device with CUDA and Tensor Core acceleration, enabling GPU-accelerated inference in a compact and energy-efficient form factor suitable for robotics applications.

4.1. Experiment Design and Evaluation Protocol

We evaluate our method on three commonly used downstream manipulation tasks, following [28, 38]: pick-and-place, sorting, and stacking. These tasks are particularly suitable because they are prevalent in SmolVLA's pretraining data. Improvements on them therefore reflect stronger underlying learning, rather than gains from introducing entirely new tasks absent from pretraining.

Performance was assessed using task-specific scoring schemes designed to capture partial and complete task execution. For the pick-and-place tasks, we employed a three-level scoring system 0, 0.5, 1, corresponding respectively to task failure, successful grasping of the cube without placement, and successful grasping followed by correct placement into the designated cup. The stacking tasks, defined as grasping one cube and placing it on top of another, were evaluated using the same discrete scale: 0 indicates failure to grasp, 0.5 indicates successful grasp without successful stacking, and 1 denotes complete execution of the stacking behavior. The sorting tasks, which require placing cubes of different colors into their assigned cups, were evaluated using a four-level scale 0.25, 0.5, 0.75, 1 to capture progressive task completion: grasping the first cube (0.25), correct placement of the first cube (0.5), correct placement of the first cube followed by grasping the second cube (0.75), and successful placement of both cubes (1). Each task was executed twice per configuration. Final task scores were aggregated across 10 repetitions and normalized by a factor of ten to yield the reported success rates.

4.2. Real World Experimental Results

Table 1 summarizes the real-world in-distribution performance across our downstream tasks. The base model corresponds to SmolVLA fine-tuned on each downstream task without any pretraining on our proposed dataset. In contrast, the remaining variants were first pretrained on 10%, 25%, 50%, or 100% of HeteroMotion and then fine-tuned on the downstream tasks under identical training conditions. Figure 2 presents the average task success rate across the three representative configurations with 95% confidence intervals. The results show that pretraining on HeteroMotion yields a clear improvement over direct fine-tuning, with the 100% pretrained model achieving a substantial performance gain compared to both the 0% and 25% variants.

Table 1. Results of real-world robotic manipulation experiments. The percentages denote the fraction of fine-tuning data employed.

Model Configuration	Task Success Rate (%)			
	Pick and Place	Stacking	Sorting	Average
SmolVLA (0% Pre-trained)	30.0	20.0	2.5	17.50
SmolVLA (10% Pre-trained)	35.0	15.0	0.0	16.67
SmolVLA (25% Pre-trained)	45.0	2.0	5.0	17.33
SmolVLA (50% Pre-trained)	**50.0**	25.0	12.5	29.17
SmolVLA (100% Pre-trained)	45.0	**30.0**	**15.0**	**30.00**

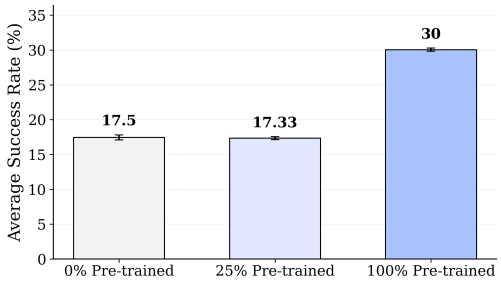


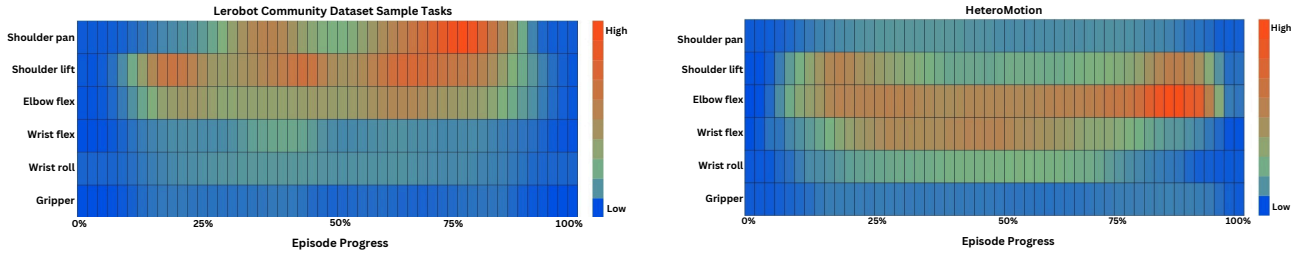
Figure 2. Average success rate across down-steram tasks. Error bars denote 95% confidence intervals.

Notably, smaller pretraining subsets (e.g., 10% and 25%) provide limited or inconsistent improvements over the base model, which is in line with previous findings that pretraining must surpass a certain threshold before yielding noticeable performance gains [42]. The 50% and especially the 100% pretraining variants show more pronounced improvements, indicating that meaningful benefits arise when sufficient data is incorporated.

The observed performance gains can be attributed to the exposure of the model during pretraining to a broader range of motions and task complexities. This broader exposure increases coverage of target distribution, rather than simply increasing sample density within a restricted subset of the action space. Domain adaptation results show that the expected target error $\epsilon_P(h)$ can be bounded as:

$$\epsilon_P(h) \leq \epsilon_Q(h) + \text{disc(Q, P)} + \lambda$$

where $\epsilon_Q(h)$ denotes the source (pretraining) error, $\text{disc}(Q, P)$ measures the divergence between the source and target distributions, and λ captures the irreducible joint error [13, 32]. Importantly, reducing $\epsilon_Q(h)$ by scaling data within a narrow region does not reduce the divergence term



(a) Lerobot Community Dataset (b) HeteroMotion

Figure 3. Comparison between the variance of Lerobot Community dataset and our proposed dataset, HeteroMotion.

disc(Q, P) if substantial portions of the target support remain uncovered. In contrast, expanding pretraining coverage reduces this distributional discrepancy. Our results are therefore consistent with improved support coverage during pretraining, rather than gains arising solely from increased data volume within a limited portion of the action space.

4.3. Joint Variance Analysis

To demonstrate the presence of motion diversity in HeteroMotion, we analyzed the variance across the six joints of the SO101. We compared samples from HeteroMotion to a merged subset of three different datasets from the LeRobot Community dataset, ensuring that they represented distinct tasks. This variance analysis is important, as prior work has suggested that low-variance regions may lack the exploratory coverage necessary for model generalization [44].

As shown in Figure 3a, the existing community dataset exhibits reasonable variance along the first three joints. This aligns with our observation of the task distribution: most pick-and-place tasks involve lifting the arm (primarily moving the shoulder lift and elbow flex joints), rotating toward the object (shoulder pan), and then moving toward a bin to place the item. The prevalence of pick-and-place behaviors is reflected in Figure 3a, where the high-variance regions showcase this movement. For example, the shoulder pan joint shows two distinct high-variance regions: one during rotation toward the object and another during rotation back toward the bin, with relatively constant motion in between.

In contrast, HeteroMotion dataset, shown in Figure 3b, exhibits sustained and more evenly distributed variance throughout the entire episode. Notably, it captures significantly greater diversity in wrist flex and wrist roll, while also maintaining variability across all joints. This broader distribution of motion better reflects diverse manipulation skills and supports our claim that increasing motion diversity is critical for scaling community-collected VLA pretraining data.

5. Conclusion

In this work, we investigated the motion diversity present in the LeRobot community dataset. Through a systematic analysis of task descriptions, we observed a strong concentration of pick-and-place tasks, suggesting limited diversity in motion patterns. We further examined this through a joint variance analysis, which further suggested that HeteroMotion dataset is dominated by a narrow set of motion primitives. To address this limitation, we collected a structured dataset spanning five behavior categories, comprising 15 tasks and over 1,050 trajectories. To evaluate the impact of increased motion diversity, we conducted real-world experiments using the baseline SmolVLA model alongside variants pretrained on different portions of our dataset. The results demonstrate clear performance improvements when larger scale pretraining is present.

Overall, our findings suggest that community-collected datasets would benefit from more intentional guidance toward complex and physically diverse tasks. Rather than focusing primarily on scaling the number of similar demonstrations, future data collection efforts should prioritize tasks that expand motion patterns, interaction complexity, and reasoning requirements. Encouraging contributors to record more compositionally and physically varied behaviors may provide a more effective path toward scalable and robust VLA models.

References

[1] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems*, 35:23716–23736, 2022. 1

[2] Jorge Aldaco, Travis Armstrong, Robert Baruch, Jeff Bingham, Sanky Chan, Kenneth Draper, Debidatta Dwibedi, Chelsea Finn, Pete Florence, Spencer Goodrich, et al. Aloha 2: An enhanced low-cost hardware for bimanual teleoperation. *arXiv preprint arXiv:2405.02292*, 2024. 1, 2

[3] Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023. 1

[4] Johan Bjorck, Fernando Castañeda, Nikita Cherniadev, Xingye Da, Runyu Ding, Linxi Fan, Yu Fang, Dieter Fox,

Fengyuan Hu, Spencer Huang, et al. Gr00t n1: An open foundation model for generalist humanoid robots. *arXiv preprint arXiv:2503.14734*, 2025. 1

[5] Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Lucy Xiaoyang Shi, James Tanner, Quan Vuong, Anna Walling, Haohuan Wang, and Ury Zhilinsky. π_0: A vision-language-action flow model for general robot control, 2026. 3

[6] Florian Bordes, Richard Yuanzhe Pang, Anurag Ajay, Alexander C Li, Adrien Bardes, Suzanne Petryk, Oscar Mañas, Zhiqiu Lin, Anas Mahmoud, Bargav Jayaraman, et al. An introduction to vision-language modeling. *arXiv preprint arXiv:2405.17247*, 2024. 1

[7] Konstantinos Bousmalis, Giulia Vezzani, Dushyant Rao, Coline Devin, Alex X Lee, Maria Bauzá, Todor Davchev, Yuxiang Zhou, Agrim Gupta, Akhil Raju, et al. Robocat: A self-improving generalist agent for robotic manipulation. *arXiv preprint arXiv:2306.11706*, 2023. 2

[8] Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Joseph Dabis, Chelsea Finn, Keerthana Gopalakrishnan, Karol Hausman, Alex Herzog, Jasmine Hsu, et al. Rt-1: Robotics transformer for real-world control at scale. *arXiv preprint arXiv:2212.06817*, 2022. 1, 2

[9] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020. 1, 2

[10] Remi Cadene, Simon Alibert, Francesco Capuano, Michel Aractingi, Adil Zouitine, Pepijn Kooijmans, Jade Choghari, Martino Russi, Caroline Pascal, Steven Palma, et al. Lerobot: An open-source library for end-to-end robot learning. In *The Fourteenth International Conference on Learning Representations*. 2

[11] Remi Cadene, Simon Alibert, Alexander Soare, Quentin Gallouedec, Adil Zouitine, Steven Palma, Pepijn Kooijmans, Michel Aractingi, Mustafa Shukor, Dana Aubakirova, Martino Russi, Francesco Capuano, Caroline Pascal, Jade Choghari, Jess Moss, and Thomas Wolf. Lerobot: State-of-the-art machine learning for real-world robotics in pytorch. https://github.com/huggingface/lerobot, 2024. 1, 3

[12] Sudeep Dasari, Frederik Ebert, Stephen Tian, Suraj Nair, Bernadette Bucher, Karl Schmeckpeper, Siddharth Singh, Sergey Levine, and Chelsea Finn. Robonet: Large-scale multi-robot learning. *arXiv preprint arXiv:1910.11215*, 2019. 1

[13] Shai Ben David, Tyler Lu, Teresa Luu, and Dávid Pál. Impossibility theorems for domain adaptation. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 129–136. JMLR Workshop and Conference Proceedings, 2010. 4

[14] Danny Driess, Fei Xia, Mehdi SM Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, et al. Palm-e: An embodied multimodal language model. *arXiv preprint arXiv:2303.03378*, 2023. 1

[15] Frederik Ebert, Yanlai Yang, Karl Schmeckpeper, Bernadette Bucher, Georgios Georgakis, Kostas Daniilidis, Chelsea Finn, and Sergey Levine. Bridge data: Boosting generalization of robotic skills with cross-domain datasets. *arXiv preprint arXiv:2109.13396*, 2021. 1

[16] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, DDL Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*, 10, 2022. 1, 2

[17] Chi-Pin Huang, Yunze Man, Zhiding Yu, Min-Hung Chen, Jan Kautz, Yu-Chiang Frank Wang, and Fu-En Yang. Fast-thinkact: Efficient vision-language-action reasoning via verbalizable latent planning. *arXiv preprint arXiv:2601.09708*, 2026. 1

[18] Physical Intelligence, Ali Amin, Raichelle Aniceto, Ashwin Balakrishna, Kevin Black, Ken Conley, Grace Connors, James Darpinian, Karan Dhabalia, Jared DiCarlo, et al. $\pi_{0.6}$: a vla that learns from experience. *arXiv preprint arXiv:2511.14759*, 2025. 2

[19] Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, et al. $\pi_{0.5}$: a vision-language-action model with open-world generalization. *arXiv preprint arXiv:2504.16054*, 2025. 1, 2

[20] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International conference on machine learning*, pages 4904–4916. PMLR, 2021. 1

[21] Peter Kairouz and H Brendan McMahan. Advances and open problems in federated learning. *Foundations and trends in machine learning*, 14(1-2):1–210, 2021. 2

[22] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020. 1

[23] Kento Kawaharazuka, Jihoon Oh, Jun Yamada, Ingmar Posner, and Yuke Zhu. Vision-language-action models for robotics: A review towards real-world applications. *IEEE Access*, 2025. 1

[24] Alexander Khazatsky, Karl Pertsch, Suraj Nair, Ashwin Balakrishna, Sudeep Dasari, Siddharth Karamcheti, Soroush Nasiriany, Mohan Kumar Srirama, Lawrence Yunliang Chen, Kirsty Ellis, et al. Droid: A large-scale in-the-wild robot manipulation dataset. *arXiv preprint arXiv:2403.12945*, 2024. 1, 2

[25] Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, et al. Openvla: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024. 1, 2

[26] Jason Lee, Jiafei Duan, Haoquan Fang, Yuquan Deng, Shuo Liu, Boyang Li, Bohan Fang, Jieyu Zhang, Yi Ru Wang, Sangho Lee, et al. Molmoact: Action reasoning models that can reason in space. *arXiv preprint arXiv:2508.07917*, 2025. 1

[27] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022. 3

[28] Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. Libero: Benchmarking knowledge transfer for lifelong robot learning, 2023. 4

[29] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023. 1

[30] Qiang Liu. Rectified flow: A marginal preserving approach to optimal transport. *arXiv preprint arXiv:2209.14577*, 2022. 3

[31] Jianlan Luo, Charles Xu, Jeffrey Wu, and Sergey Levine. Precise and dexterous robotic manipulation via human-in-the-loop reinforcement learning. *Science Robotics*, 10(105): eads5033, 2025. 3

[32] Yishay Mansour, Mehryar Mohri, and Afshin Rostamizadeh. Domain adaptation: Learning bounds and algorithms. *arXiv preprint arXiv:0902.3430*, 2009. 4

[33] NVIDIA Corporation. Nvidia jetson orin nano super developer kit, 2025. Accessed: 2026-02. 4

[34] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 1

[35] Abby O'Neill, Abdul Rehman, Abhiram Maddukuri, Abhishek Gupta, Abhishek Padalkar, Abraham Lee, Acorn Pooley, Agrim Gupta, Ajay Mandlekar, Ajinkya Jain, et al. Open x-embodiment: Robotic learning datasets and rt-x models: Open x-embodiment collaboration 0. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6892–6903. IEEE, 2024. 1, 2

[36] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021. 1, 2

[37] Scott Reed, Konrad Zolna, Emilio Parisotto, Sergio Gomez Colmenarejo, Alexander Novikov, Gabriel Barth-Maron, Mai Gimenez, Yury Sulsky, Jackie Kay, Jost Tobias Springenberg, et al. A generalist agent. *arXiv preprint arXiv:2205.06175*, 2022. 1

[38] Mustafa Shukor, Dana Aubakirova, Francesco Capuano, Pepijn Kooijmans, Steven Palma, Adil Zouitine, Michel Aractingi, Caroline Pascal, Martino Russi, Andres Marafioti, et al. Smolvla: A vision-language-action model for affordable and efficient robotics. *arXiv preprint arXiv:2506.01844*, 2025. 1, 2, 3, 4

[39] Octo Model Team, Dibya Ghosh, Homer Walke, Karl Pertsch, Kevin Black, Oier Mees, Sudeep Dasari, Joey Hejna, Tobias Kreiman, Charles Xu, et al. Octo: An open-source generalist robot policy. *arXiv preprint arXiv:2405.12213*, 2024. 1

[40] Homer Rich Walke, Kevin Black, Tony Z Zhao, Quan Vuong, Chongyi Zheng, Philippe Hansen-Estruch, Andre Wang He, Vivek Myers, Moo Jin Kim, Max Du, et al. Bridgedata v2: A dataset for robot learning at scale. In *Conference on Robot Learning*, pages 1723–1736. PMLR, 2023. 2

[41] Yixiao Wang. Generalization capability for imitation learning. *arXiv preprint arXiv:2504.18538*, 2025. 2

[42] Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, et al. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*, 2022. 4

[43] Yusong Wu, Ke Chen, Tianyu Zhang, Yuchen Hui, Taylor Berg-Kirkpatrick, and Shlomo Dubnov. Large-scale contrastive language-audio pretraining with feature fusion and keyword-to-caption augmentation. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2023. 1

[44] Thomas T. Zhang, Daniel Pfrommer, Chaoyi Pan, Nikolai Matni, and Max Simchowitz. Action chunking and exploratory data collection yield exponential improvements in behavior cloning for continuous control, 2025. 5

[45] Tony Z Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning fine-grained bimanual manipulation with low-cost hardware. *arXiv preprint arXiv:2304.13705*, 2023. 1, 3

Appendix

A. Additional Dataset Details

Table 2. Overview of the 15 manipulation tasks by category.

Category	Task	Description
Conditional Manipulation	• If the block is near the sides, move it inward; else move it toward the sides.	• Requires conditional decision-making based on spatial proximity to table boundaries.
	• If the object is closer to the bin than the box, move it to the box; else move it to the bin.	• Involves comparative spatial reasoning between two target containers.
	• If the object is behind the bowl, pull it forward; else push it behind the bowl	• Tests relational spatial understanding based on a reference object.
Spatial Arrangement	• Place the marker between two blocks	• Evaluates precise spatial placement within a constrained configuration.
	• Arrange blocks in a straight horizontal line	• Requires structured alignment across multiple objects.
	• Place the block to the right of the marker	• Assesses directional spatial reasoning relative to a fixed reference.
Non-Grasp Manipulation	• Slide the block to the right side of the table	• Tests non-prehensile manipulation without lifting or grasping.
	• Push the block into the marked region	• Requires accurate planar control to move the object into a predefined region.
	• Push the block until it touches the box	• Evaluates contact-aware motion execution and physical interaction.
Exclusion Tasks	• Put all blocks except the red one into the bin	• Tests selective reasoning by filtering objects by attribute.
	• Move everything except the block to	• Requires identifying and preserving a designated object while relocating others.
	• Move every non blue object to the bin	• Evaluates attribute-based filtering and selective object transfer.
Human Interaction Tasks	• Pick the block and give it to the human	• Assesses coordinated grasping and safe object handover.
	• Take the block from the human hand and put it into the bin	• Requires bidirectional human-robot interaction and placement.
	• Pick the cracker and give it to the human	• Evaluates safe manipulation of fragile objects during interaction.