2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

MOVE: A Simple Motion-Based Data Collection Paradigm for Spatial Generalization in Robotic Manipulation

Huanqian Wang[1,3*†], Chi Bene Chen[1*], Yang Yue[1*§], Danhua Tao[4], Tong Guo[1], Shaoxuan Xie[2]
Denghang Huang[2], Shiji Song[1], Guocai Yao[2‡], Gao Huang[1‡]

***Abstract*— Imitation learning method has shown immense promise for robotic manipulation, yet its practical deployment is fundamentally constrained by the data scarcity. Despite prior work on collecting large-scale datasets, there still remains a significant gap to robust spatial generalization. We identify a key limitation: individual trajectories, regardless of their length, are typically collected from a *single, static spatial configuration* of the environment. This includes fixed object and target spatial positions as well as unchanging camera viewpoints, which significantly restricts the diversity of spatial information available for learning. To address this critical bottleneck in data efficiency, we propose MOtion-Based Variability Enhancement (*MOVE*), a simple yet effective data collection paradigm that enables the acquisition of richer spatial information from dynamic demonstrations. Our core contribution is an augmentation strategy that injects motion into any movable objects within the environment for each demonstration. This process implicitly generates a dense and diverse set of spatial configurations within a single trajectory. We conduct extensive experiments in both simulation and real-world environments to validate our approach. For example, in simulation tasks requiring strong spatial generalization, *MOVE* achieves an average success rate of 39.1%, a 76.1% relative improvement over the static data collection paradigm (22.2%), and yields up to 2–5× gains in data efficiency on certain tasks. Our code is available at https://github.com/lucywang720/MOVE[5].**

I. Introduction

Recently, end-to-end learning methods have made significant strides in robotic control, enabling the completion of numerous complex manipulation tasks. The state-of-the-art approaches, exemplified by Diffusion Policy [1, 2] and Vision-Language-Action models [3–5], leverage large-scale datasets to achieve impressive generalization capabilities across different objects, new tasks, and varying environments [6, 7]. These advancements mark a significant step towards general-purpose embodied intelligence. Despite these achievements, generalization across spatial variations in object pose remains a critical yet overlooked challenge [8, 9]. This limitation is particularly acute for real-world deployment, where robots must operate in unstructured environments far more variable than the controlled settings in simulation environments.

The root of this problem is the inefficient sampling of spatial configuration from a continuous state space by static data collection methods. We highlight the limitations of static data collection in Figure 1 left. We trained a diffusion policy using data uniformly gathered from 9 spatial locations and evaluated the policy across the entire object space. The policy, as expected, succeeds only around the locations in the training set, but fails at other test points in much of the remaining space, resulting in a success rate of 29.5%. This issue becomes especially severe as the spatial dimensionality of the task space increases. For instance, diverse camera perspectives, adjustable table heights, and randomized target object placements contribute to a more complex and combinatorially rich spatial setting. As shown in Table I, success rates decay exponentially as the spatial dimensions expand, revealing the poor generalization capabilities of current methods in real-world environments.

To tackle this problem, we challenge the paradigm of static data collection itself. In this paradigm, an entire expert trajectory, often spanning several hundred timesteps, captures a task under a fixed spatial configuration, such as a fixed object position, target position, and camera viewpoint. Spatial sparsity leads to the consequence that if the policy needs to grasp an object in a new pose, completely new demonstrations must be collected for that specific location [10, 11], which is impractical for real-world deployment.

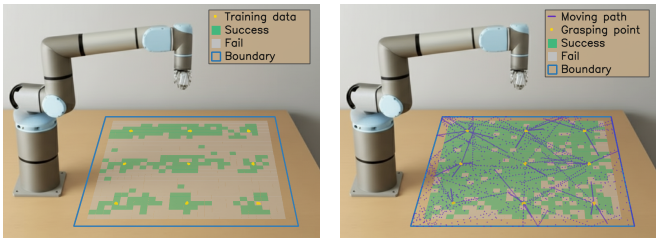

Fig. 1: We uniformly sample 10 trajectories from each of the 9 points across the entire space using both static data collection and MOVE. To ensure a fair comparison, we enforce that the grasping point of each MOVE trajectory corresponds to that of a static trajectory and the same total number of timesteps. Despite this alignment, MOVE exhibits significantly better spatial generalization to unseen grasp points (29.5% vs. 80.8%).

In this work, we introduce *MOVE*, a motion-based data collection framework which enhances the spatial information density per trajectory to improve the spatial generalization in robotic manipulation. Motivated by limitations of traditional data collection, we aim to endow a single trajectory with spatial location information from more than just one spatial configuration. Specifically, the key objects, such as pickup object, target object and camera, are intentionally and contin-

*Equal contribution. §Project lead. ‡Corresponding author.

[1]BNRist, Tsinghua University. [2]Beijing Academy of Artificial Intelligence. [3]Anyverse Dynamics. [4]Southeast University. [5]The real-world dataset is available at https://huggingface.co/datasets/BAAI/MOVE.

†Work partially done during an internship in Anyverse Dynamics.

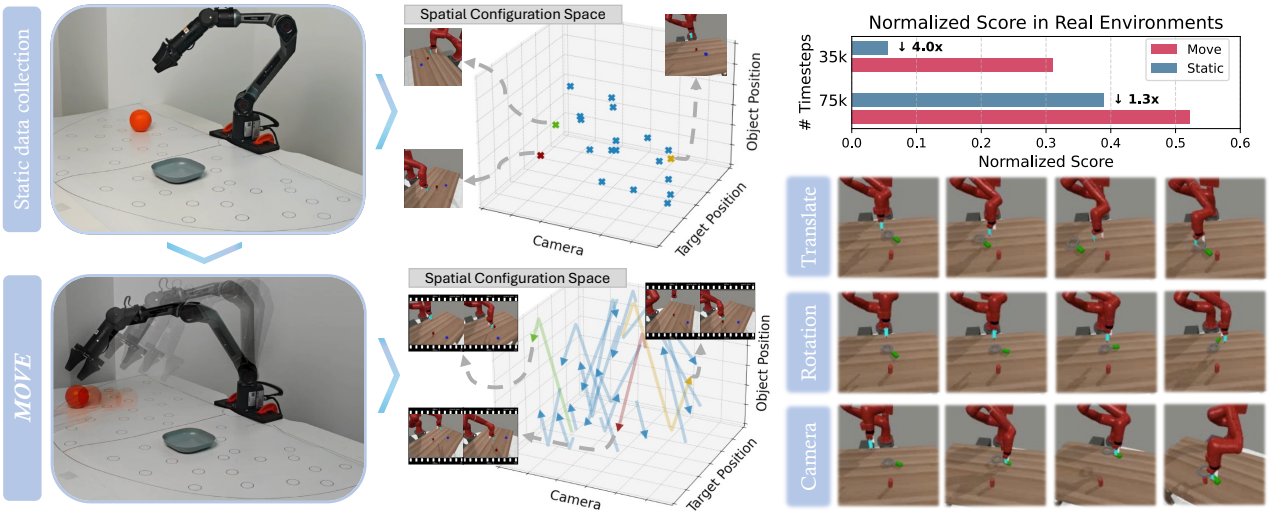

Fig. 2: **An overview of the MOVE data collection paradigm.** (**Left**) A conceptual comparison between the standard static data collection paradigm and *MOVE*. The former samples from discrete, fixed spatial configurations, where each trajectory represents a single point in the spatial configuration space. In contrast, each trajectory collected by *MOVE* is treated as a continuous segment, with objects, targets, and the camera in motion, resulting in a dense and diverse set of spatial configurations within a single trajectory. Therefore, with the same number of trajectories, our approach encodes broader spatial coverage and richer spatial information. (**Right Top**) In real-world environments, policies trained with data collected via *MOVE* demonstrate superior performance and generalization compared to the traditional data collection paradigm, with a maximum improvement of 4.0x on the normalized score. (**Right Bottom**) We demonstrate several forms of motion augmentation employed in *MOVE*, including translation, rotation and camera motion.

uously moved when collecting expert human demonstrations. We illustrate the spatial coverage of *MOVE* compared to static methods in Figure 2. Given an equivalent dataset volume, *MOVE* exposes the policy to a stream of continuously moving objects within a single trajectory during training. Although the policy may not directly learn to grasp objects at every position along the motion path, it can still implicitly acquire knowledge of the corresponding spatial configurations. As shown in Figure 1 (right), a policy trained with *MOVE* data is able to grasp the object along its motion path. This paradigm effectively embeds a powerful and flexible form of data augmentation directly into each trajectory, thereby enhancing both sample efficiency and spatial generalization.

We validate the effectiveness of *MOVE* through extensive experiments in both simulation and real-world scenarios and demonstrate that a policy trained with *MOVE* data collection paradigm evidently improves data efficiency and achieves better spatial generalization. In the Meta-World simulation environments requiring strong spatial generalization, *MOVE* on average achieves a success rate of 39.1%, a 76.1% improvement over the static data collection paradigm (22.2%). For example, in the Pick-and-Place task, a 20k-timestep *MOVE* dataset matches the success rate of a 50k-timestep static dataset, and in the Assembly task, a 50k-timestep *MOVE* dataset achieves comparable performance to a 100k static dataset. Furthermore, in real-world experiments with highly randomized spatial configurations that pose a substantial generalization challenge, *MOVE* achieves a 23.3% success rate with only 35k timesteps, dramatically outperforming the static method under the same data budget (3.3% at 35k) and matching its performance of the static method that is trained with more than twice the data (23.3% at 75k). These results highlight that *MOVE* substantially reduces the amount of data required to achieve the same level of spatial generalization, demonstrating dynamic data collection paradigm's potential to enable spatial generalization more effectively and efficiently that supports scalable learning.

II. Related Works

A. Robotic manipulation and spatial generalization

Robot manipulation has recently made significant progress, where policies represented by Vision-Language-Action (VLA) models and diffusion models enable robots to perform a wide range of tasks based on visual inputs. Built upon Vision-Language Models, VLA models utilize large pretrained transformers to map visual and linguistic inputs to robotic actions. [3–5, 12–14]. Diffusion Policy [1] and other extensions [2, 15–18], leverage the capabilities of diffusion models to fit multi-modal action distributions and enhance long-horizon planning and efficiency by using action chunking, demonstrated remarkable success in learning complex, dexterous skills directly from human demonstrations.

Despite these advances, achieving spatial generalization is still a central and long-term challenge in robotics. Many prior works attempted to solve this problem by improving visual representations by injecting spatial information such as 3D point clouds or bounding box [11, 19–21]. Nevertheless, the performance of robotic policies remains critical to the scale and diversity of training datasets [6, 22], and degrades severely especially when the test scenarios are different from the training distribution. As a result of this dependency, the field is shifting from a purely model-centric to a data-centric viewpoint [23, 24], and researchers are dedicated to addressing the challenge that collecting large-scale, real-world robotic data is resource-intensive and time-consuming.

B. Data Collection in Robotics

Inspired by the success of data scaling in LLMs and VLMs, researchers have also begun exploring data scaling for manipulation tasks. The development of large-scale datasets has been instrumental to recent progress, including DROID (76k trajectories) [24], BridgeData V2 (∼60k trajectories) [25] and the Open X-Embodiment dataset (∼1M trajectories) [26]. Despite significant efforts from the community, the quantity of available robot data remains far below that of vision-language data, limiting the ability of current methods to achieve robust generalization [27]. To address this challenge, recent research investigated more efficient data acquisition methods, following directions including large-scale physical simulation and 3D scene reconstruction. Simulation-based methods collect extensive data in high-fidelity simulations to bridge the sim-to-real gap [28–31]. 3D reconstruction-based methods can generate synthetic trajectories based on real trajectories [10, 32]. Beyond these methods, researchers are also exploring efficient strategies for collecting real-world data. ADC [33] shares some similarities with our work, as it periodically resets the object's position during data collection. However, ADC includes only a few discrete points along a trajectory, whereas *MOVE* captures richer spatial information by forming a continuous curve in the location space. Moreover, *MOVE* naturally extends to additional spatial dimensions, allowing the incorporation of variables such as camera motion and dynamic table height.

III. Approach

A. Challenge in Spatial Generalization

In robot learning, the generalization of policy training heavily relies on large, diverse datasets, but acquiring enough data in robotics is notably difficult to achieve. This challenge becomes more pronounced as the spatial dimensionality of the task increases, since each standard demonstration typically contains only a single instance of a spatial configuration, leading to severe spatial sparsity in high-dimensional environments. We validate this phenomenon on the Meta-world Pick-Place task and list the results in Table I.

Specifically, we construct three settings of increasing difficulty for spatial generalization by progressively randomizing key spatial factors. In setting 1, Only the object's position is randomly initialized within a $30\,\text{cm} \times 30\,\text{cm}$ area on the table, while the target position and camera viewpoint remain fixed. Although training and testing use the same sampling range, generalization is required due to the difference in sampled positions. In setting 2, in addition to randomizing the object's position, the target position is also randomized within a $20\,\text{cm} \times 10\,\text{cm} \times 25\,\text{cm}$ volume. Setting 3 further increases difficulty by randomizing the camera pose, with the viewing angle ranging from 0 to π radians, transitioning from a controlled setup to a more realistic scenario. Under the same training budget of 20k timesteps, the success rate drops dramatically from 67.5% in Setting 1 to 31.7% in Setting 3. This exponential decline highlights the inability of the standard static data collection paradigm to sufficiently cover the spatial configuration space in realistic environments with multi-dimensional variation.

Overview. To enhance the model's generalization to complex environments and improve the robustness against unforeseen spatial positions, we explore a simple yet effective data collection approach termed **MOtion-Based Variability Enhancement** (*MOVE*), which leverages dynamic trajectories to provide richer spatial grounding signals. As Figure 2 shows, to increase the coverage of spatial configurations, we introduce controlled kinematic motions, including translation, rotation, and camera movement, as a data augmentation strategy when collecting training data (Section III-B). Once collected, we apply diffusion policy to train policy models (Section III-C).

TABLE I: **Impact of Real-World Spatial Variation on Success Rate.** We progressively introduce variations in object placement, target placement, and camera viewpoint. Unlike simulation settings where these factors are typically fixed or only slightly perturbed, each training and testing trajectory is collected under *a randomized configuration* sampled from the same distribution.

Randomized Factors	From research setting $\Longrightarrow$ real world		
	Object	+ Target	+ Camera
Success rate	0.675	0.447	0.317

B. Spatial Configuration Augmentation

Object Translation: To ensure spatial coverage of the entire workspace, *MOVE* simulates multiple linear motion trajectories for *both the pickup and target objects*, which is modelled by incorporating linear translation and bounce at the boundaries. An object's position $\mathbf{p}_i(t)$ at time t is determined by its initial position $\mathbf{p}_i(0)$, a constant velocity vector v_i, and moving direction $\mathbf{d}_i$.

$$\begin{aligned} \mathbf{p}_i(t) &= \mathbf{p}_i(0) + t \cdot v_i \cdot v_{\max} \cdot \mathbf{d}_i, \quad \forall i \in \{\text{pick}, \text{target}\} \\ \text{with} \quad & v_i \sim \mathrm{B}(\alpha_p, \beta_p), \quad \mathbf{d}_i \sim U(\mathbb{S}^2) \end{aligned} \tag{1}$$

where $v_{\max}$ is the maximum possible speed and velocity v_i is sampled from a Beta distribution $\mathrm{B}(\alpha_p, \beta_p)$. We leverage Beta distribution's properties to constrain the sampled velocity to the interval $[0, v_{\max}]$ and ensure a higher probability of speeds approaching 0 than $v_{\max}$. Specifically, we set all $\alpha = 2$ and $\beta = 5$ throughout our simulation experiments. This distribution facilitates the model's learning of not only spatial generalization but also robust grasping strategies.

Object Rotation: To improve spatial generalization to a wide range of object orientations, we introduce constant angular velocity rotations. For simplicity, we model 1-D rotation around the vertical z-axis. Similarly to the above, the orientation $\theta_i(t)$ of an object evolves based on its initial orientation $\theta_i(0)$ and a constant angular velocity ω_i.

$$\begin{aligned} \boldsymbol{\theta}_i(t) &= \boldsymbol{\theta}_i(0) + t \cdot \omega_i \cdot \omega_{\max} \cdot \mathbf{d}_i, \quad \forall i \in \{\text{pick}, \text{target}\} \\ \text{with} \quad & \omega_i \sim \mathrm{B}(\alpha_\theta, \beta_\theta), \quad \mathbf{d}_i \in \{-1, 1\} \end{aligned} \tag{2}$$

This augmentation is particularly beneficial for asymmetric objects such as mugs with handles. In contrast, it is not applied to objects that are fully rotationally symmetric.

Camera Movement: To simulate a non-static viewpoint, the virtual camera moves along a constrained cylindrical path relative to the scene's center. The camera's position is updated similarly to object translation, with its velocity sampled from a Beta distribution.

$$\begin{aligned} \mathbf{p}_i(t) = \mathbf{p}_i(0) + t \cdot u_i \cdot u_{\max} \cdot \mathbf{d}_i, \quad \forall i \in \{\text{camera}\} \\ \text{with} \quad u_i \sim \mathrm{B}(\alpha_c, \beta_c), \quad \mathbf{d}_i \sim U(\mathbb{S}^2) \end{aligned} \tag{3}$$

Combined Augmentation Strategy: Rather than applying all motions simultaneously, we employ a staged strategy tailored to the semantic phases of a task. For example, in the Box-Close task, we decompose each trajectory into the pick phase ($t_0 \to t_1$) and a placement phase ($t_1 \to t_2$).

- Pick phase ($t_0 \to t_1$): we apply translation and rotation only to the pickup object (the box lid), the camera movement is also introduced. This forces the policy to learn to approach and grasp a moving target while adapting to a changing viewpoint.
- Placement phase ($t_1 \to t_2$): we apply linear translation only to the target object (the box body) and continue the dynamic camera motion. This challenges the policy to place the object onto a moving destination.

This strategy expands *MOVE* beyond a single dimension, increasing the spatial information richness across multiple spatial dimensions. We validate this combined augmentation strategy in Section IV-E.

C. Training

To learn the robot's control strategy, we adopt the Diffusion Policy [1] framework. For training, we utilize the dynamic dataset collected following the methodology described in Section III-B. Specifically, we employ the Denoising Diffusion Implicit Models (DDIM) scheduler for a deterministic and efficient sampling process.The less noisy sample x_{t-1} is computed from the noised sample x_t at timestep t as follows:

$$\begin{aligned} \mathbf{x}_{t-1} = \sqrt{\bar{\alpha}_{t-1}} \left(\frac{\mathbf{x}_t - \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)}{\sqrt{\bar{\alpha}_t}} \right) \\ + \sqrt{1-\bar{\alpha}_{t-1}} \cdot \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t) \end{aligned} \tag{4}$$

A comprehensive list of hyperparameters, architecture details, and other settings can be referred to in the Appendix A.

IV. Experiments

A. Preliminary Experiment

We begin with preliminary experiments designed to showcase the effectiveness of *MOVE* in achieving better spatial generalization under a clean and controlled setting in the Pick-Place simulation task. Specifically, we define a set of discrete grasping points and ensure that both the static data collection method and *MOVE* perform grasps at these same locations during data collection. In the *MOVE* setting, objects are initialized at random positions and then move toward the predefined grasping points. We also keep the total number of timesteps for training the same across both methods.

a) Generalization Comparison from Sparse Sampling: We uniformly choose 9 points across the space. For each point, we sample 10 trajectories for static data collection as empirically sparse sampling with less trajectories usually failed to learn a meaningful policy. We sampled total 85 trajectories for *MOVE* to match the total timesteps. We evaluate both policies across the entire space and plot the results in Figure 1. Although the policy trained on static data performs well on the training points, *MOVE* demonstrates superior generalization to the entire space, succeeding even in locations far from training points. Specifically, *MOVE* achieves a global success rate of 80.8% across the entire space, significantly outperforming the static policy's 29.5%.

b) Generalization Comparison from Dense Sampling: While *MOVE* demonstrates strong performance in sparse sampling scenarios, this sampling strategy can be inefficient. Therefore, we validate our method under a dense sampling strategy, which provides better spatial coverage for a fixed dataset size than concentrating on few points. We uniformly sample 90 (static) and 85 (*MOVE*) trajectories, ensuring that both datasets have the same total number of timesteps. We then evaluate across the entire space and visualize the results in Figure 3. Although the dense static sampling baseline naturally achieves a higher success rate, *MOVE* still yields substantial improvements from 66% to 74%, especially in regions of the state space that are otherwise difficult to learn.

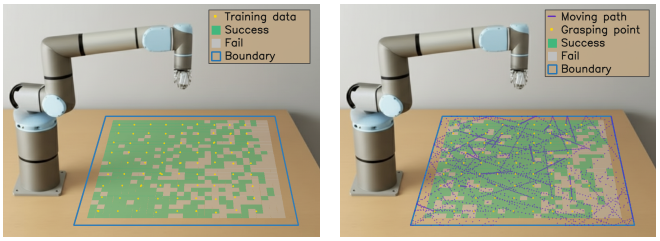

Fig. 3: **Generalization Comparison from Dense Sampling**. For fair comparison, we enforce the grasp point of each MOVE trajectory corresponds to that of a static trajectory. Despite training on the same set of grasping positions, MOVE exhibits better generalization to unseen grasp points (66% vs. 74%).

c) Generalization Comparison from Circle Sampling: To investigate the influence of data sampling location on *MOVE*'s performance, we sample grasping points evenly distributed on a circle and constrain the object's motion path within this circle. We sample 90 (static) and 59 (*MOVE*) trajectories, ensuring that both datasets have the same total number of timesteps. We evaluate both policies across the entire space and visualize the results in Figure 5. Surprisingly, *MOVE* not only dominates within the sampling circle but also significantly outperforms the baseline in the outer regions. Specifically, *MOVE* achieves a success rate of 43.7% and outperforms the static policy's 21.3% across the in-circle space; *MOVE* also achieves a success rate of 67.4% while the static policy achieves 44.6% across the out-of-circle space.

B. Experiment Setup

The following experiments are designed to comprehensively evaluate the effectiveness of *MOVE*, and compare

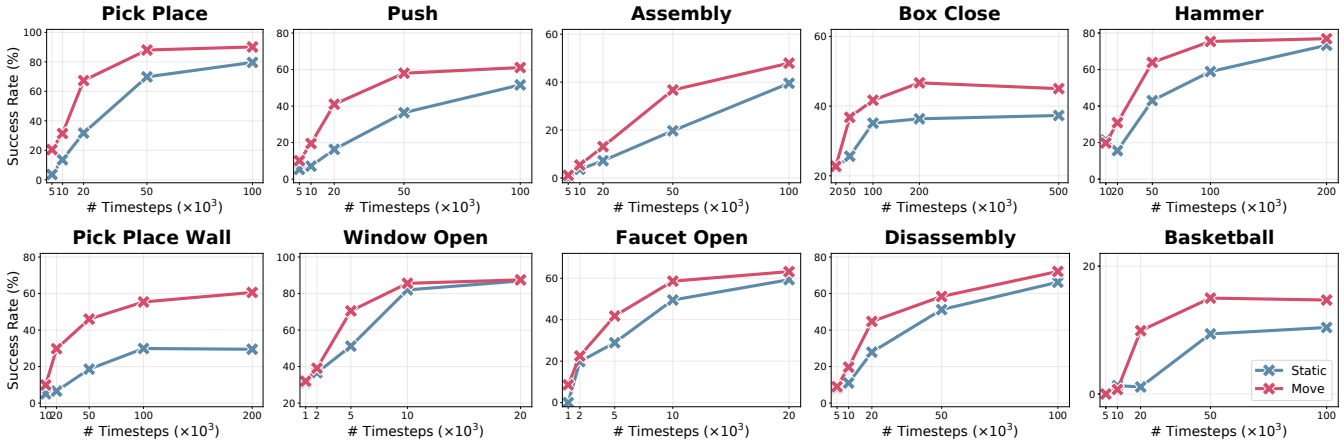

Fig. 4: **Efficient scaling with demonstrations.** Success rate across 10 simulation tasks. Specifically, the x-axis represents the number of timesteps, where each timestep corresponds to a single robot action, rather than the number of trajectories. *MOVE* consistently outperforms the static data collection paradigm at each data scaling point.

TABLE II: **Main results from the Meta-World simulation environment.** We *simulate real-world spatial generalization challenges* by testing each time under randomized spatial configurations (object and target positions, camera view points, *et, al*). We rerun both data collection and training processes with 3 distinct random seeds and report the average success rate. We ensure all methods use the same total number of timesteps, and thus require nearly the same amount of human effort. The amount of training data is provided in Table V.

Method	Pick Place	Push	Assembly	Box Close	Hammer	Pick Place Wall	Window Open	Faucet Open	Disassembly	Basketball	Average
Static	0.317	0.163	0.072	0.351	0.156	0.066	0.512	0.289	0.279	0.011	0.222
ADC [33]	0.472	0.335	0.053	0.383	0.252	0.237	0.455	0.345	0.207	0.025	0.276
MOVE	**0.673**	**0.410**	**0.131**	**0.417**	**0.309**	**0.298**	**0.705**	**0.418**	**0.447**	**0.099**	**0.391** (↑ **76.1**%)

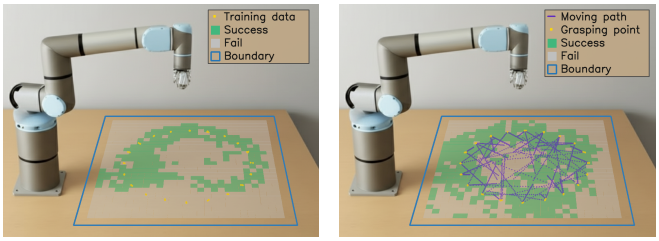

Fig. 5: We uniformly sample grasping points evenly distributed on a circle and constrain the object's motion path within this circle for *MOVE*. While being exposed to the same grasping positions and constrained within the circle, MOVE exhibits significantly better spatial generalization on both the in-circle region (21% vs. 44%) and the out-of-circle region (45% vs. 67%).

this method with the traditional static paradigm. We conduct experiments in both simulation and real-world environments.

a) Tasks and Environments: In simulation environments, we leverage the Meta-World benchmark [34] which comprises a series of robotic manipulation tasks including grasping, pushing and placement. For each simulation environment, we implement random initialization of the object, target, and camera over wide ranges by modifying the simulator code. We select a specific number of environments from each of three difficulty levels as representative examples due to time-costly code modification.

For real-world validation, we use a canonical pick-and-place task. An Agilex PIPER arm is tasked with grasping an orange from a variable initial position and placing it onto a plastic tray. The operational workspace is configured as a semicircle, covering an $0.5 \times \pi \times 60$ cm $\times 60$ cm $\approx 5655\text{cm}^2$ area, which encompasses the majority of the end-effector's reachable space. The training data set was constructed by randomly sampling 20 pairs of positions, where each pair specifies one orange location and one plate location. More details are provided in the appendix B.

b) Baselines: To demonstrate the benefits of our approach, we compare *MOVE* against the conventional static data collection paradigm. Furthermore, we compare against the ADC method [33], which periodically resets the object's position to a new random location during a single data collection trajectory to encourage policy diversity. For all methods, we use the same training and evaluation protocols.

c) Expert Data Collection: In simulation, expert demonstrations are generated using the scripted policies provided by Meta-World. In the real world, we collect human demonstrations by teleoperating the robot arm using the Pika gripper. A key consideration is that trajectories collected under dynamic conditions are often longer than static ones (see Table VI). To ensure a fair comparison of data efficiency, we define the dataset size by the total number of environment interaction steps rather than the number of trajectories.

C. Results in Simulated Environments

We first evaluate *MOVE* on MetaWorld. Performance in 10 simulation tasks with varying demonstration sizes is visualized in Figure 4. Detailed average success rates are presented in Table II. The results illustrate that *MOVE* consistently and remarkably outperforms the static data collection paradigm across all tasks and dataset sizes, specifically improving the

success rate by 76.1% with equal data. In the Pick-Place-Wall task, *MOVE* achieves comparable performance on the 20k-sized dynamic dataset to the 100k-sized static dataset, demonstrating a data efficiency of up to 5x.

D. Results in Real Environments

In this subsection, we transition the experiments to a real-world environment to validate the effectiveness of our method. For model training, we collected datasets comprising 35k and 75k timsteps, respectively. For evaluation, we test the learned policy on a grid of 30 initial object positions that were unseen during training, sampled from a 40cm $\times$ 80 cm workspace located within the semicircle. The detailed results are presented in Table III. Notably, when the dataset size is 35k, the performance of *MOVE* is nearly comparable to that of the static data collection method using a 75k dataset.

TABLE III: **Main results from the real environment.** We report the success rate and normalized score across different dataset sizes. Following prior work [22], we divide the task into three steps: approaching the correct grasping pose, picking up the orange, and putting it on the plate. Each successful step is scored 1 point. We report a normalized score, defined as Normalized score $= \frac{\text{Total test score}}{3 \times \text{Number of steps}}$, with a maximum value of 1.

# Timesteps	Method	Success Rate	Normalized Score
35k	static	3.3%	0.055
	MOVE	**23.3%**	**0.311**
75k	static	23.3%	0.389
	MOVE	**36.7%**	**0.522**

E. Ablation Study

To validate our design choices, we conduct ablation studies on two representative tasks: Pick-Place and Assembly.

a) The Impact of Dynamic Dimension Combination: We hypothesize that combining multiple dynamic spatial dimensions, such as pickup object position, target object position, and camera viewpoint, within a single trajectory enriches spatial information and is critical for learning a robust and generalizable policy. Therefore, *MOVE* employs combinations of dynamic dimensions during data collection to maximize spatial diversity.

To validate this, we follow the same setup as in the main experiments, but selectively apply *MOVE* to individual spatial dimensions while keeping the others static. The results, shown in Figure 6, highlight the impact of combinations of dynamic dimensions. For example, in the Pick-Place task, we start with object translation as the initial dynamic component, then incrementally add target object translation, and finally incorporate camera motion. The results demonstrate that as more dynamic dimensions are introduced in *MOVE*, the success rate consistently improves.

b) The Impact of Augmentation Hyperparameters: The maximum velocities, $v_{\max}$ (translation), $\omega_{\max}$ (rotation), and $u_{\max}$ (camera motion), serve as hyperparameters that control the intensity of spatial augmentation (see Section III-B). To evaluate their impact, we select $v_{\max}$ at varying speed levels and present results in Figure 7. The results show that our method exhibits a degree of robustness to speed variations.

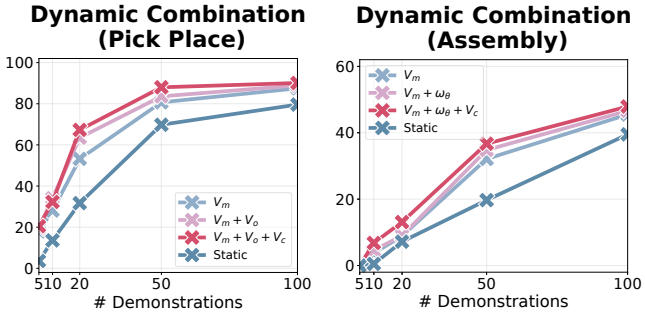

Fig. 6: **The Impact of Dynamic Dimension Combination.** V_m refers to *MOVE* applied only to the pickup object's motion. $+V_o$ indicates the addition of target object motion, $+V_c$ further incorporates dynamic camera movement, $+\omega_\theta$ denotes dynamic object rotation.

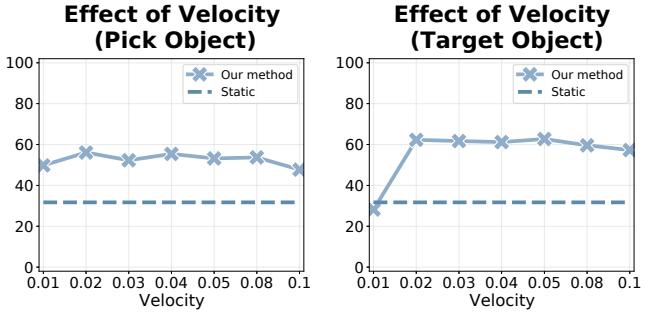

Fig. 7: The impact of $v_{\max}$ on success rate for the pickup object (left) and the target object (right).

V. Conclusion and Limitation

In this paper, we introduce a simple yet effective data collection paradigm that significantly enhances spatial generalization in robotic manipulation and alleviate the challenge of data scarcity. By incorporating dynamic spatial configurations into demonstrations, *MOVE* provide much richer spatial information in each trajectory. This is evidenced by *MOVE*'s consistent improvements over static data collection across varying dataset scales. While our study demonstrates promising results, it also has several limitations. For instance, it remains to be explored how *MOVE* can be applied in real-world environments to spatial variations such as changing camera viewpoints, as well as to more complex manipulation tasks like folding garments. We expect this limitation to be mitigated by developing mechanical devices that can adjust camera viewpoints and introduce perturbations such as translations, shuffling, and rotations to garments. We leave these directions to future work.

VI. Acknowledgement

This work is supported in part by the National Key Research and Development Program of China under Grant 2022ZD0114903, the National Natural Science Foundation of China under Grants U24B20173, and the Scientific Research Innovation Capability Support Project for Young Faculty under Grant ZYGXQNJSKYCXNLZCXM-I20. The backgrounds in Figure 1, 3, 5 are generated with Gemini [35], while task-related data and annotations are not.

References

[1] C. Chi, Z. Xu, S. Feng, E. Cousineau, Y. Du, B. Burchfiel, *et al.*, "Diffusion policy: Visuomotor policy learning via action diffusion," *Robotics: Science and Systems*, 2023.

[2] S. Liu, L. Wu, B. Li, H. Tan, H. Chen, Z. Wang, *et al.*, "RDT-1b: a diffusion foundation model for bimanual manipulation," in *The Thirteenth International Conference on Learning Representations*, 2025. [Online]. Available: https://openreview.net/forum?id=yAzN4tz7oI

[3] Octo Model Team, D. Ghosh, H. Walke, K. Pertsch, K. Black, O. Mees, *et al.*, "Octo: An open-source generalist robot policy," in *Proceedings of Robotics: Science and Systems*, 2024.

[4] C.-L. Cheang, G. Chen, Y. Jing, T. Kong, H. Li, Y. Li, *et al.*, "Gr-2: A generative video-language-action model with web-scale knowledge for robot manipulation," 2024, https://arxiv.org/abs/2410.06158.

[5] K. Black, N. Brown, D. Driess, A. Esmail, M. Equi, C. Finn, *et al.*, "π_0: A vision-language-action flow model for general robot control," 2024, https://arxiv.org/abs/2410.24164.

[6] A. Xie, L. Lee, T. Xiao, and C. Finn, "Decomposing the generalization gap in imitation learning for visual robotic manipulation," in *ICRA*, 2024, pp. 3153–3160.

[7] E. Teoh, S. Patidar, X. Ma, and S. James, "Green screen augmentation enables scene generalisation in robotic manipulation," *arXiv preprint arXiv:2407.07868*, 2024.

[8] E. Xing, A. Gupta, S. Powers, and V. Dean, "Kitchenshift: Evaluating zero-shot generalization of imitation-based policy learning under domain shifts," in *NeurIPS 2021 Workshop on Distribution Shifts: Connecting Methods and Applications*, 2021.

[9] N. Tsagkas, A. Sochopoulos, D. Danier, S. Vijayakumar, C. X. Lu, and O. M. Aodha, "When pre-trained visual representations fall short: Limitations in visuomotor robot learning," 2025, https://arxiv.org/abs/2502.03270.

[10] Z. Xue, S. Deng, Z. Chen, Y. Wang, Z. Yuan, and H. Xu, "Demogen: Synthetic demonstration generation for data-efficient visuomotor policy learning," 2025, https://arxiv.org/abs/2502.16932.

[11] H. Tan, X. Xu, C. Ying, X. Mao, S. Liu, X. Zhang, *et al.*, "Manibox: Enhancing spatial grasping generalization via scalable simulation data generation," 2024, https://arxiv.org/abs/2411.01850.

[12] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, X. Chen, K. Choromanski, *et al.*, "Rt-2: Vision-language-action models transfer web knowledge to robotic control," 2023, https://arxiv.org/abs/2307.15818.

[13] M. J. Kim, K. Pertsch, S. Karamcheti, T. Xiao, A. Balakrishna, S. Nair, *et al.*, "Openvla: An open-source vision-language-action model," 2024, https://arxiv.org/abs/2406.09246.

[14] Y. Yue, Y. Wang, B. Kang, Y. Han, S. Wang, S. Song, *et al.*, "Deer-vla: Dynamic inference of multimodal large language models for efficient robot execution," *NeurIPS*, vol. 37, pp. 56 619–56 643, 2024.

[15] Y. Ze, G. Zhang, K. Zhang, C. Hu, M. Wang, and H. Xu, "3d diffusion policy: Generalizable visuomotor policy learning via simple 3d representations," 2024, https://arxiv.org/abs/2403.03954.

[16] T.-W. Ke, N. Gkanatsios, and K. Fragkiadaki, "3d diffuser actor: Policy diffusion with 3d scene representations," *arXiv preprint arXiv:2402.10885*, 2024.

[17] J. Wen, Y. Zhu, J. Li, Z. Tang, C. Shen, and F. Feng, "Dexvla: Vision-language model with plug-in diffusion expert for general robot control," *arXiv preprint arXiv:2502.05855*, 2025.

[18] X. Ma, S. Patidar, I. Haughton, and S. James, "Hierarchical diffusion policy for kinematics-aware multi-task robotic manipulation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 18 081–18 090.

[19] H. Zhu, H. Yang, Y. Wang, J. Yang, L. Wang, and T. He, "Spa: 3d spatial-awareness enables effective embodied representation," *arXiv preprint arXiv:2410.08208*, 2024.

[20] D. Qu, H. Song, Q. Chen, Y. Yao, X. Ye, Y. Ding, *et al.*, "Spatialvla: Exploring spatial representations for visual-language-action model," 2025, https://arxiv.org/abs/2501.15830.

[21] X. Li, P. Li, M. Liu, D. Wang, J. Liu, B. Kang, *et al.*, "Towards generalist robot policies: What matters in building vision-language-action models," 2024, https://arxiv.org/abs/2412.14058.

[22] F. Lin, Y. Hu, P. Sheng, C. Wen, J. You, and Y. Gao, "Data scaling laws in imitation learning for robotic manipulation," 2025, https://arxiv.org/abs/2410.18647.

[23] Q. Bu, J. Cai, L. Chen, X. Cui, Y. Ding, S. Feng, *et al.*, "Agibot world colosseo: A large-scale manipulation platform for scalable and intelligent embodied systems," *arXiv preprint arXiv:2503.06669*, 2025.

[24] A. Khazatsky, K. Pertsch, S. Nair, A. Balakrishna, S. Dasari, S. Karamcheti, *et al.*, "Droid: A large-scale in-the-wild robot manipulation dataset," 2025, https://arxiv.org/abs/2403.12945.

[25] H. Walke, K. Black, A. Lee, M. J. Kim, M. Du, C. Zheng, *et al.*, "Bridgedata v2: A dataset for robot learning at scale," 2024, https://arxiv.org/abs/2308.12952.

[26] E. Collaboration, A. O'Neill, A. Rehman, A. Gupta, A. Maddukuri, A. Gupta, *et al.*, "Open x-embodiment: Robotic learning datasets and rt-x models," 2025, https://arxiv.org/abs/2310.08864.

[27] Y. Zhong, F. Bai, S. Cai, X. Huang, Z. Chen, X. Zhang, *et al.*, "A survey on vision-language-action models: An action tokenization perspective," 2025, https://arxiv.org/abs/2507.01925.

[28] S. Zhu, G. Wang, X. Kong, D. Kong, and H. Wang, "3d gaussian splatting in robotics: A survey," 2024, https:

//arxiv.org/abs/2410.12262.
[29] Y. Mu, T. Chen, S. Peng, Z. Chen, Z. Gao, Y. Zou, *et al.*, "Robotwin: Dual-arm robot benchmark with generative digital twins (early version)," 2025, https://arxiv.org/abs/2409.02920.
[30] T. Mu, Z. Ling, F. Xiang, D. Yang, X. Li, S. Tao, *et al.*, "Maniskill: Generalizable manipulation skill benchmark with large-scale demonstrations," 2021, https://arxiv.org/abs/2107.14483.
[31] T.-D. Do, N. Gireesh, J. Wang, and H. Wang, "Watch less, feel more: Sim-to-real rl for generalizable articulated object manipulation via motion adaptation and impedance control," 2025, https://arxiv.org/abs/2502.14457.
[32] A. Mandlekar, S. Nasiriany, B. Wen, I. Akinola, Y. Narang, L. Fan, *et al.*, "Mimicgen: A data generation system for scalable robot learning using human demonstrations," 2023, https://arxiv.org/abs/2310.17596.
[33] S. Huang, Y. Liao, S. Feng, S. Jiang, S. Liu, H. Li, *et al.*, "Adversarial data collection: Human-collaborative perturbations for efficient and robust robotic imitation learning," 2025, https://arxiv.org/abs/2503.11646.
[34] T. Yu, D. Quillen, Z. He, R. Julian, A. Narayan, H. Shively, *et al.*, "Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning," 2021, https://arxiv.org/abs/1910.10897.
[35] G. Team, R. Anil, S. Borgeaud, J.-B. Alayrac, J. Yu, R. Soricut, *et al.*, "Gemini: A family of highly capable multimodal models," 2025. [Online]. Available: https://arxiv.org/abs/2312.11805

APPENDIX

A. Training Details

TABLE IV: Training Details of diffusion policy. We used different training steps in simulation and real environments.

Hyper-parameters	Simulation	Real-world
Action Prediction Horizon	4	8
Action Horizon	3	8
Observation Horizon	2	2
Gradient Step / epoch	220000	200
batch size	128	128
Train Denoise Step	100	50
Val Denoise Step	10	10

TABLE V: The results presented in Table II primarily use a dataset of 20k timesteps. For particularly simple or challenging tasks, smaller (5k) or larger (100k) datasets are used accordingly.

Task	Pick	Push	Box	Assembly	Hammer
# Timesteps	20k	20k	100k	20k	20k
Task	Wall	Window	Faucet	Disassembly	Basketball
# Timesteps	20k	5k	5k	20k	20k

B. Data collection and Scoring Details

a) Data Collection: In the static data collection setup, Operator 1 performs teleoperation by Pika teleoperation device while Operator 2 positions the objects manually. In contrast, the *MOVE* paradigm retains the same teleoperation procedure but extends Operator 2's role to include dynamic object manipulation. Specifically, Operator 2 employs a 3D-printed transparent resin tongs to move oranges or plates during each trajectory. Note that Operator 2 remains outside the camera's field of view to avoid interfering with the visual input used for policy training. A visual overview of the real-world data collection process is shown in Figure 9.

TABLE VI: Comparison of static and *MOVE* collection paradigm.

Metrics	Real-world		Simulation	
	Average timesteps	Personnel Required	Average timesteps	Generation successful rate
Static	459.2	2	83.8	93.3%
MOVE	549.3	2	107.4	89.7%

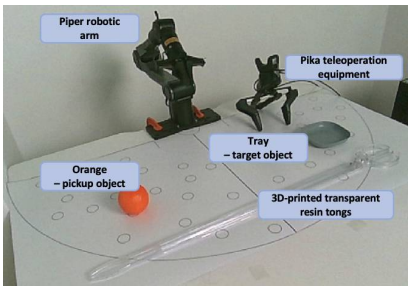

Fig. 8: An overview of our real-world experimental setup, captured by a RealSense camera. We sample pairs of random points on the table as placement locations for the oranges and trays, which serve as training data. Object movements are performed using 3D-printed transparent resin tongs.

Fig. 9: An example of the *MOVE* data collection paradigm. Captured using an external smartphone for better visibility, rather than the camera used for data collection.

b) Scoring Criteria: Following [22], we report the normalized score as a metric for the robot manipulation to concretely evaluate the capability of the policy step by step.

- 0 points: The gripper does not move toward the orange or moves around it without any contact.
- 1 point: The gripper touches the orange but does not grasp it due to minor errors.
- 2 points: The gripper successfully grasp the orange.
- 3 points: The gripper put the orange on the tray.