

InternData-A1: Pioneering High-Fidelity Synthetic Data for Pre-training Generalist Policy

Yang Tian^{*,1,2}, Yuyin Yang^{*,1}, Yiman Xie^{*,1}, Zetao Cai^{*,1}, Xu Shi^{*,1}, Ning Gao¹, Hangxu Liu¹, Xuekun Jiang¹, Zherui Qiu¹, Feng Yuan¹, Yaping Li¹, Ping Wang², Junhao Cai¹, Jia Zeng^{†,1}, Hao Dong^{†,2} and Jiangmiao Pang^{†,1}

¹Shanghai AI Laboratory, ²Peking University

Recent works explore how real and synthetic data contribute to Vision-Language-Action (VLA) models' generalization. While current VLA models have shown the strong effectiveness of large-scale real-robot pre-training, synthetic data has not previously demonstrated comparable capability at scale. This paper provides the first evidence that synthetic data alone can match the performance of the strongest π -dataset in pre-training a VLA model, revealing the substantial value of large-scale simulation. The resulting model also exhibits surprisingly zero-shot sim-to-real transfer on several challenging tasks. Our synthetic dataset, InternData-A1, contains over 630k trajectories and 7,433 hours across 4 embodiments, 18 skills, 70 tasks, and 227 scenes, covering rigid, articulated, deformable, and fluid-object manipulation. It is generated through a highly autonomous, fully decoupled, and compositional simulation pipeline that enables long-horizon skill composition, flexible task assembly, and heterogeneous embodiments with minimal manual tuning. Using the same architecture as π_0 , we pre-train a model entirely on InternData-A1 and find that it matches the official π_0 across 49 simulation tasks, 5 real-world tasks, and 4 long-horizon dexterous tasks. We release the dataset and will open-source the generation pipeline to broaden access to large-scale robotic data and to lower the barrier to scalable data creation for embodied AI research.

 [Data: InternData-A1](#) |  [Homepage](#)



Figure 1. InternData-A1 pioneers a large-scale, high-fidelity synthetic dataset with physically faithful, photorealistic rendering, diverse object domains (rigid, articulated, fluid, and deformable), extensive multi-skill tasks, and broad cross-embodiment coverage.

* equal contributions, † equal corresponding authors

© 2025 Shanghai Artificial Intelligence Laboratory. All rights reserved.

Dataset	Traj.	Skill	Task	Scene	Embodiment	Fluid	Deformable	Failure Recovery	Open-Source	Collection
MimicGen (Mandlekar et al., 2023)	50k	-	18	1	4	x	x	x	✓	Teleoperation & Augmentation
ManiSkill2 (Gu et al., 2023)	30k	20	20	-	1	x	x	x	✓	TAMP & MPC & RL
RLBench (James et al., 2020)	2.7k	-	100	1	1	x	x	x	✓	Waypoints & Motion Planners
LIBERO (Liu et al., 2023)	5k	-	130	20	1	x	x	x	✓	Teleoperation
CALVIN (Mees et al., 2022)	37.5k	-	34	4	1	x	x	x	✓	Teleoperation
RoboCasa (Nasiriany et al., 2024)	77k	8	100	120	2	x	x	x	✓	Teleoperation & Augmentation
GraspVLA (Deng et al., 2024)	10M	1	1	1	1	x	x	x	x	Autonomous
RoboTwin 2.0 (Chen et al., 2025a)	100k	-	50	1	5	x	x	x	✓	Autonomous
InternVLA-M1 (Chen et al., 2025b)	244k	2	1	1	1	x	x	x	✓	Autonomous
InternData-A1	630k	18	70	227	4	✓	✓	✓	✓	Autonomous

Table 1. Comparison to existing simulation datasets.

1. Introduction

Manipulation lies at the core of embodied intelligence, enabling robots to assist in daily tasks and deliver real economic value. Recent advancements (Bjorck et al., 2025; Black et al., 2024; Bu et al., 2025a; Intelligence et al., 2025; Li et al., 2025) demonstrate that large-scale real-robot data can endow Vision-Language-Action (VLA) models with strong generalization across tasks, scenes, and embodiments. However, collecting real data at such scale is resource-intensive: teleoperation demands skilled operators, specialized hardware, and extensive human labor, making large, diverse real-world datasets infeasible for most research groups. As a result, the broader community lacks the ability to systematically study the data requirements that underpin successful VLA pre-training.

Simulation offers a promising complementary avenue: its rich asset libraries, controllable scenes, and automated data generation pipelines create the possibility of scaling manipulation data far beyond what is practical with real robots. Yet existing simulated datasets (Chen et al., 2025a,b; Deng et al., 2024; Mandlekar et al., 2023; Nasiriany et al., 2024) still cover narrow skill sets (primarily pick-and-place), focus mainly on rigid objects, require nontrivial human operation, and rarely validate their effectiveness for large-scale VLA pre-training. This leaves an open question: can high-fidelity synthetic data, when scaled sufficiently in embodiments, scenes, skills, and physical realism, match the pre-training effectiveness of the strongest real-world datasets?

As depicted in fig. 1 and table 1 , we introduce InternData-A1, a high-fidelity synthetic manipulation dataset comprising 630k trajectories and 7,433 hours across 4 embodiments, 18 skills, 70 tasks, and 227 scenes, covering rigid, articulated, deformable, and fluid-object interactions. InternData-A1 is generated through a fully decoupled, autonomous simulation pipeline that separates asset specification, skill policies, task composition, and rendering. Each task is built by retrieving embodiments, scenes, and objects from an asset library, then composing scripted skill policies that compute and interpolate trajectories conditioned on robot and object states into complete behaviors. This compositional design flexibly supports bimanual manipulation, multi-robot coordination, and extended multi-stage tasks, while automatically generating object-level annotations, domain randomization, and collision-aware motion plans. With framework optimizations, the pipeline produces 209.7 hours robot data per day on 8 RTX 4090 GPUs with minimal manual tuning and a cost below 0.003 US Dollars per episode, enabling scalable synthesis of physically and visually faithful demonstrations. This pipeline underpins the broad task diversity of InternData-A1 and enables systematic VLA validation through large-scale pre-training and sim-to-real evaluation.

We demonstrate that a π_0 model pre-trained exclusively on InternData-A1 achieves comparable performance to the official π_0 trained on the closed-source π -dataset across diverse real-world scenarios. This result, for the first time, establishes that large simulation-only data can match the strongest real-world data for VLA pre-training. Across 49 simulation, 5 real-world, and 4 long-horizon dexterous tasks, the synthetic-pretrained model closely matches the official π_0 model. Furthermore, compared with existing open-source datasets (Bu et al., 2025a; Nasiriany et al., 2024; O’Neill et al.,

2024), models pre-trained on InternData-A1 consistently outperform counterparts in both simulated and real-world evaluations, highlighting the strength of InternData-A1 for VLA pre-training.

In addition, ten selected simulated tasks achieve direct sim-to-real transfer with an average success rate exceeding 50%. Further analysis on four representative tasks reveals that, under well-aligned sim-to-real settings, fewer than 1,600 simulated samples can match the performance of 200 real samples, highlighting the high fidelity of InternData-A1 and its minimal sim-to-real gap.

In conclusion, while the π -dataset remains closed-source, InternData-A1 and its generation pipeline are open-sourced and reproducible in simulation. We hope this work offers an accessible supplement to embodied AI research and enables a deeper understanding of data scaling effects in robot learning.

2. Related Works

Large-scale Robot Datasets. Existing large-scale robot datasets fall into two categories: simulation-based (Gu et al., 2023; James et al., 2020; Mees et al., 2022) and real-world datasets (Jiang et al., 2025; Walke et al., 2023). Simulation enables controllable variation in embodiments, scenes, and objects, and supports both teleoperation-based augmentation (Nasiriany et al., 2024) and automated pipelines (Chen et al., 2025b; Deng et al., 2024). However, teleoperation still requires substantial manual effort, and most automated systems mainly target picking tasks. RoboTwin 2.0 (Chen et al., 2025a) recently expanded simulation to bimanual manipulation with cross-embodiment diversity and domain randomization. Real-world datasets (Black et al., 2024; Bu et al., 2025a; Khazatsky et al., 2024; O’Neill et al., 2024; Wu et al., 2024) offer high-fidelity demonstrations without simulation gaps, but are costly to scale and typically exhibit narrow embodiment coverage, limited scene diversity, or variable data quality. The π -series datasets (Black et al., 2024) demonstrate that large-scale real-robot pre-training can yield strong generalization across tasks, scenes, and embodiments, highlighting the importance of scale and diversity for VLA models. In contrast to prior simulation datasets, InternData-A1 provides broad task diversity, heterogeneous embodiments, photorealistic rendering, and long-horizon multi-skill trajectories, enabling systematic study of large-scale synthetic data pre-training for VLA models.

VLA Models. The generalization capability of VLA models are primarily determined by the scale, diversity, and source of their pre-training data. Existing approaches fall into three broad categories. (1) Real-data-only models (Black et al., 2024; Brohan et al., 2022; Cheang et al., 2025, 2024; Kim et al., 2024; Yang et al., 2025b; Zitkovich et al., 2023) achieve strong in-domain performance (Black et al., 2024) but remain tied to specific training environments, often exhibiting limited open-world generalization (Bu et al., 2025b; Tian et al., 2025). (2) Simulation-only models (Deng et al., 2024) benefit from massive synthetic trajectories and extensive domain randomization, but typically cover narrow skill families (e.g., grasping), constraining their applicability. (3) Hybrid models combine simulation and real demonstrations (Bjorck et al., 2025; Chen et al., 2025b) or incorporate web-scale VQA data (Intelligence et al., 2025; Qu et al., 2025; Yang et al., 2025a; Zhai et al., 2025) to improve grounding and long-horizon planning. In this work, we leverage InternData-A1 to provide VLA models with stronger action priors and generalization ability, achieving downstream performance on par with state-of-the-art real data despite the inherent physics and visual gaps of simulation. Accordingly, we adopt the π_0 model and its underlying π -dataset as our primary baselines.

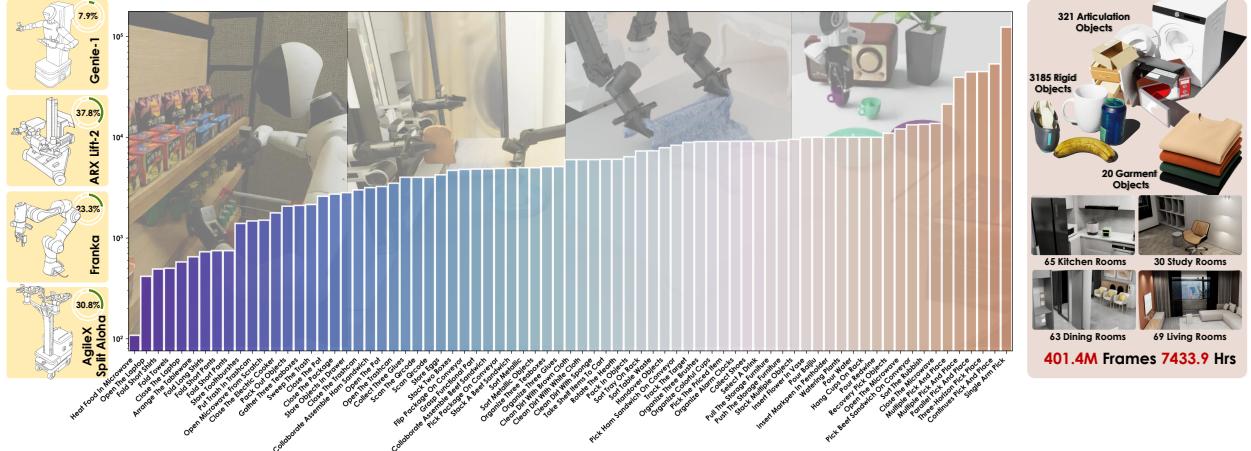


Figure 2. **Data Statistics.** InternData-A1 provides 4 single or dual-arm embodiments, 70 diverse tasks, 3185 rigid objects, 321 articulation objects, 20 garments, and 227 rooms. All these elements consist 630k episodes, 401.4M frames and 7433.9 hours.

3. Data Statistics

In this section, we elaborate on the data statistics from three aspects: embodiments, tasks, and assets, with the overall scale and diversity visualized in fig. 2.

Embodiments. InternData-A1 includes four types of robotic embodiments: **Agibot Genie-1**, one of the most widely used dual-arm robots in large-scale real robot data collection factories (Bu et al., 2025a), accounts for 7.9%. **Franka Emika Panda**, one of the most common single-arm manipulators in research laboratories, constitutes 23.3%. We fix Franka on a table to perform tabletop manipulation tasks, including both regular and long-horizon scenarios. **AgileX Split Aloha** equipped with Piper-100 arms, and **ARX Lift-2** equipped with R5a arms, are two popular embodiments in current VLA (Black et al., 2024; Intelligence et al., 2025; Zheng et al., 2025) real-world deployments, occupying 30.8% and 37.8%, respectively. These dual-arm systems enable rich bimanual manipulation, including sequential skills (e.g., make a sandwich) and parallel behaviors (e.g., fold shirts) across diverse scenes.

Tasks. To ensure both trajectory-level and task-level diversity, InternData-A1 is constructed from a comprehensive set of manipulation primitives spanning nearly all fundamental human-like skills—from folding and pouring to rotating and stacking. By composing these primitives, we design 70 diverse tasks across realistic scenarios, including 4 fluid-related, 4 deformable-object, 15 articulated, and 47 rigid-object tasks. Notably, unlike prior works (Chen et al., 2025a; James et al., 2020; Mees et al., 2022; Nasiriany et al., 2024), our tasks are not defined by simply varying manipulated objects (e.g., picking hundreds of objects still counts as one task in our statistics); each task instead specifies a distinct context, composition of atomic skills, and action-space constraint. InternData-A1 also includes 18 long-horizon tasks, each involving at least three sequential skills, totaling 124,789 trajectories and 141,421,619 frames. Beyond diversity, we ensured balance by proportionally mixing data from different tasks, resulting in 56 of the 70 tasks containing between 1,000 and 10,000 trajectories, approaching a near-uniform task distribution.

Assets. We provide **3,185 rigid objects** from 107 categories, sourced from OmniObject3D (Wu et al., 2023) and Objaverse (Deitke et al., 2023). We aggregate articulated objects from multiple sources, including GRUtopia (Wang et al., 2024), GAPartNet (Geng et al., 2023), GenSim2 (Hua et al., 2024), Infinite Mobility (Lian et al., 2025), and ArtVIP (Jin et al., 2025). In total, this results in **321 articulated objects** across 14 categories, ranging from small items such as cartons to large household

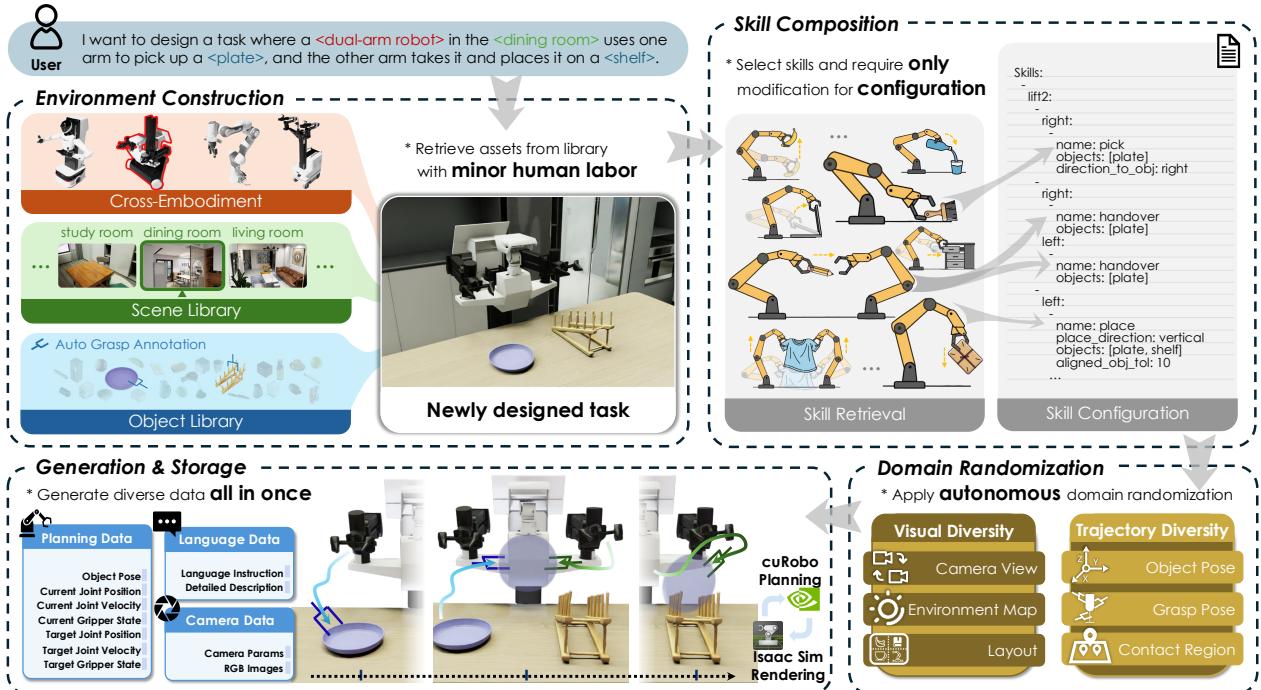


Figure 3. **Data synthesis pipeline of InternData-A1.** It consists of four stages: (1) environment construction with selected embodiments, scenes, and objects; (2) task composition using modular atomic skills invoked via simple configuration commands; (3) domain randomization over layouts, object poses, lighting, etc.; and (4) trajectory generation, where CuRobo (Sundaralingam et al., 2023) interpolates dense joint actions, validates them through physics simulation, and renders only successful trajectories into the LeRobot format.

appliances like refrigerators. For garment assets, we use the EinScan Rigel Pro scanner to digitize **20 real garments** (short-sleeve shirts, sweaters, shorts, and towels). Regarding the environments, InternData-A1 contains **227 indoor scenes** (kitchens, study rooms, dining rooms, and living rooms), forming a diverse and realistic layout setup. All these scenes are acquired and segmented from the GRScenes-100 dataset in GRUtopia (Wang et al., 2024).

4. Data Synthesis

In this section, we outline the data synthesis pipeline: we first describe environment construction (robots, scenes, and objects) (Sec 4.1), then detail how skills constitute complete tasks (Sec 4.2), and finally explain the domain randomization (Sec. 4.3), generation, and storage procedures (Sec. 4.4).

4.1. Environment Construction

As illustrated in fig. 3, given a task description template, we first retrieve task-relevant assets from the library, including robots, scenes, and objects. The robot asset is a stable USD-type embodiment that has been verified for consistent contact dynamics in robot–object interactions. The scene asset is a room-level environment, selected and segmented from GRUtopia (Wang et al., 2024). Each scene is annotated with detailed manipulation-area metadata, enabling natural and context-aware task construction. Our object library spans **rigid**, **articulated**, **deformable**, and **fluid** categories. Rigid objects include physical attributes, canonical poses, and automatically generated grasp poses following AnyGrasp (Fang et al., 2023). Articulated objects are annotated with accurate joint axes, part poses,

and physical parameters (e.g., damping, stiffness) for stable interaction. Deformable assets are remeshed garments with material and contact properties, simulated using Vertex Block Descent (Chen et al., 2024) for realistic behavior. Fluid objects are modeled with particle-based dynamics, where particles are adaptively generated within containers, and liquid surfaces reconstructed via isosurface rendering with diverse PBD materials. All functional annotations are managed through a unified interface, enabling easy asset configuration with high physical fidelity.

4.2. Skill Composition

Once the environments are prepared, users compose tasks by selecting atomic skills from the skill library. Each skill is a modular scripted policy that takes object states (poses, joint states), robot states (base and end-effector poses), and user-defined constraints as inputs, and outputs a sequence of waypoints (target end-effector 6D poses). Waypoints serve as a unified representation, cleanly decoupling high-level skill logic from low-level motion execution. For example, the *Pick* skill filters grasp candidates for the target object and computes the pre-grasp, grasp, and post-grasp poses, while the articulation-related *Push* skill uses contact annotations to determine pre-contact, contact, and post-contact waypoints. For tasks with special spatial constraints, we provide a set of constraint options for each skill. For instance, in the task *Insert Flower In Vase*, it requires the stem to remain upright. Users could specify the displacement and rotation between the placed object and container in *Place* skill. All these well-engineered skills can be invoked through simple command templates in a configuration file (see fig. 3). Since each skill is designed as an automatic state-waypoint mapping, no additional cost is required when varying objects, spatial ranges, scenes or even embodiments. The only manual effort required is merely the adjustment of spatial ranges. Users simply specify which arm (left or right) to use for each skill and organize them sequentially or in parallel for bimanual tasks, and then a long-horizon task unfolds smoothly and systematically.

4.3. Domain Randomization

To enrich visual diversity, we perturb the primary and wrist camera views within $\pm 5^\circ$ rotations and ± 5 cm translations. We construct a library of 174 environment maps, each with randomized light temperature and intensity to simulate diverse natural illumination conditions. Target objects can be replaced with others from the same category, while the tabletop and background layouts are also randomized. To further enhance trajectory diversity, object positions and orientations are sampled within task-specific spatial ranges. For manipulation functions, we introduce additional randomness to the contact regions. For instance, our autonomous grasp pose generation pipeline produces millions of grasp candidates; after filtering, the final grasp pose is randomly selected from the top 40 high-confidence candidates following Anygrasp (Fang et al., 2023). For articulated and deformable objects, such as microwaves or garments, we expand the contact region into a neighborhood area and sample the contact points randomly.

4.4. Generation & Storage

After all preparations, we employ the CuRobo motion planner (Sundaralingam et al., 2023) to interpolate dense joint-space actions between the waypoints generated by each skill. For each complete robot episode, we record object metadata, language instructions, multi-view RGB images and camera parameters, as well as robot proprioceptive states and action control labels. Users can also easily store additional information, such as depth maps, grounding annotations, and bounding boxes, with simple configuration options during recording. All final data are converted into the standard LeRobot format for VLA pre-training.

4.5. Framework Optimization

In traditional synthetic data generation pipelines, trajectory planning and visual rendering are integrated into a single stage. While this architecture is suitable for rapid development and iteration, it exhibits substantial efficiency bottlenecks when scaled to large-scale data generation. The root causes can be summarized as follows:

1. **Declining planning success rate with increasing task complexity.** As task complexity grows, the success rate of trajectory planning decreases significantly. Failed trajectories do not require subsequent visual rendering, yet the single-stage architecture incurs redundant rendering overhead, resulting in unnecessary computational waste.
2. **Mismatch in computational characteristics.** Trajectory planning is fundamentally CPU-bound and executed serially, whereas visual rendering relies on GPU-based parallel computation. Executing these heterogeneous workloads in a serial manner leads to poor overall hardware utilization.

To mitigate these bottlenecks, we introduce a multi-level system optimization at the framework level. Our design includes:

1. **Stage decoupling with a pipelined architecture:** Trajectory planning and visual rendering are decoupled into two independent stages, with a pipelined execution mechanism established between the Planner and Renderer.
2. **Dynamic resource scheduling:** To address heterogeneous time-cost ratios across different tasks, we incorporate parallel batch processing strategies within both the Planner and Renderer, together with a dynamic scheduling algorithm to maximize resource utilization.
3. **Rendering efficiency optimization:** We introduce a stacked rendering (Stack Render) technique to further increase rendering throughput.
4. **Cluster stability mechanisms:** To handle stability issues and load imbalance in large-scale cluster deployments, we design a *Balancer* module for load distribution and a *Supervisor* module for monitoring and control, jointly improving cluster utilization and system robustness.

With these optimizations, our pipeline achieves a **2–3×** end-to-end performance improvement over the baseline. It further supports long-duration stable operation and efficient large-scale synthetic data generation, substantially improving productivity in synthetic data production.

5. Pre-training Effects

In this section, we address the central question of whether InternData-A1 can match the real-robot dataset in its capacity to pre-train VLA models.

5.1. Comparison with π -dataset

Experimental Setup. We benchmark against the strongest real-robot data, π -dataset. We adopt the same architecture as π_0 model, which consists of a vision-language model (Paligemma (Beyer et al., 2024)) and a flow-matching-based action expert. We pre-train a new π_0 model from Paligemma weights and a randomly initialized action expert, using solely InternData-A1, and benchmark it against the official π_0 checkpoint by fine-tuning and evaluating on downstream tasks. This comparison reflects the impact of pre-training data quality and provides the fairest available solution, as π -dataset remains fully closed-source. For simulation evaluation, we use 49 bimanual tasks from

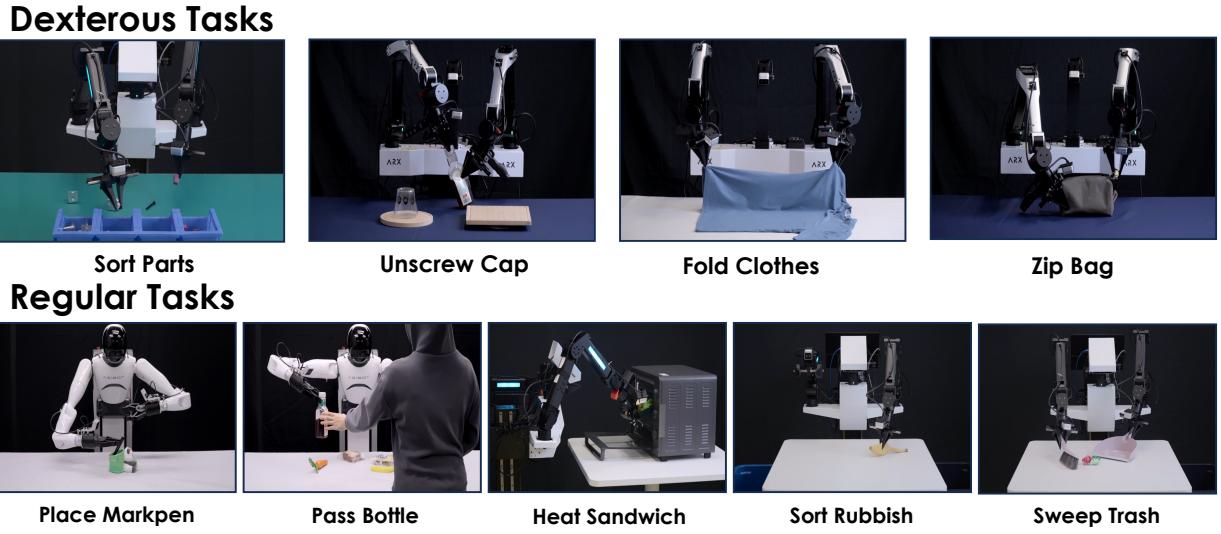


Figure 4. **Real-world setup.** We evaluate the model pre-trained on InternData-A1 against the π -dataset baseline on a suite of nine real-world tasks spanning three robots.

RoboTwin 2.0 (Chen et al., 2025a) under two difficulty modes: *Easy* (clean) and *Hard* (cluttered). We report average success rates over 100 trials across two checkpoints. For real-world evaluation, we test on three embodiments: Genie-1, ARX Lift-2 and ARX AC One. We design five regular and four dexterous tasks covering articulated manipulation, garment folding, contact-rich control, precision grasping, and long-horizon sequences. Each task is evaluated over 30 trials. In the following results, we use π_0 (*Scratch*) to denote the model without being pre-trained, π_0 for the official checkpoint, and π_0 (*InternData-A1*) for the model trained exclusively on our dataset.

Method	Hanging Mug		Lift Pot		Pick Dual Bottles		Place Object Stand		Shake Bottle Horizontally		Turn Switch		Avg. (49 Tasks)	
	Easy	Hard	Easy	Hard	Easy	Hard	Easy	Hard	Easy	Hard	Easy	Hard	Easy	Hard
π_0 (<i>Scratch</i>)	5.0%	2.0%	26.5%	0.0%	1.5%	0.5%	9.0%	0.0%	55.0%	2.0%	9.0%	9.5%	23.5%	2.5%
π_0	11.0%	6.5%	17.0%	1.5%	58.0%	16.0%	43.0%	14.0%	96.5%	55.0%	27.5%	30.0%	55.0%	20.0%
π_0 (<i>InternData-A1</i>)	24.5%	20.0%	63.5%	2.5%	62.0%	19.0%	48.5%	29.5%	98.0%	64.0%	40.5%	32.5%	60.0%	26.5%

Table 2. **Comparison to π -dataset in simulations.** π_0 model pre-trained on InternData-A1 outperforms the official π_0 model by 5% and 6.5% across 49 tasks in easy and hard settings respectively, showing the effectiveness of InternData-A1.

Evaluating in Simulation. section 5.1 summarizes the results over 49 tasks and 19,600 rollouts. Under an identical post-training process, models pre-trained on InternData-A1 consistently outperform those trained on the π -dataset, achieving 6% higher success rates in the *Easy* mode and 5% higher in the *Hard* mode. The improvement in hard settings (clean expert demonstrations and domain-randomized evaluation) indicates that the robustness to visual and spatial variations gained from InternData-A1’s extensive domain randomization persists even when downstream fine-tuning uses only clean, non-randomized data. Moreover, since InternData-A1 consists of diverse tasks freely composed from repeated atomic skills across varied contexts, we believe it provides a transferable and broader action prior for downstream tasks sharing similar skill components. This is evidenced by the six representative tasks in section 5.1, which demand robust combinations of pick, place, move, handover, and articulation-related skills. Overall, InternData-A1 improves over *Paligemma* by 36.5%

in *Easy* and 24.0% in *Hard* mode, demonstrating the strong pre-training effectiveness of InternData-A1.

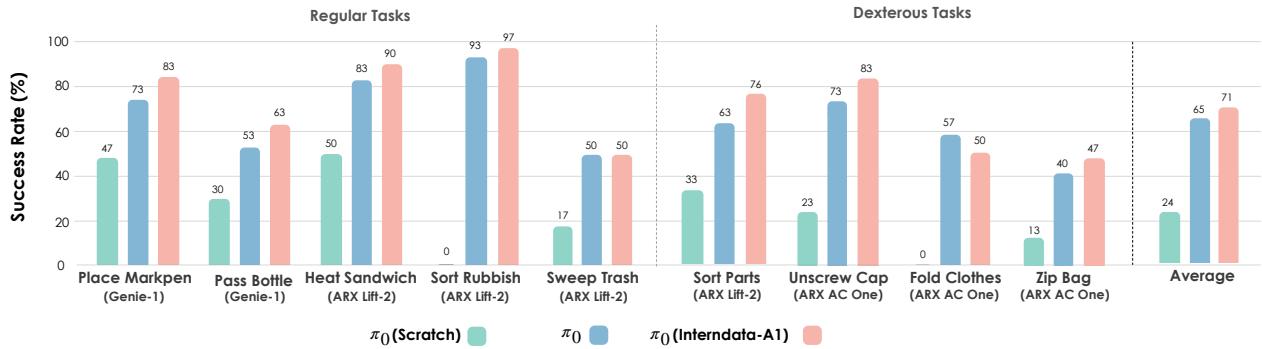


Figure 5. **Comparison to π -dataset in real-world tasks.** InternData-A1 achieves performance comparable to π -dataset across 9 real-world tasks, including 4 dexterous ones, demonstrating its strong pre-training capability.

Evaluating over Regular Tasks. We first compare InternData-A1 and π -dataset on five representative real-world base tasks across two embodiments, covering articulation (*Heat Sandwich*), basic pick-and-place (*Sort Rubbish*, *Place Markpen*), human–robot interaction (*Pass Bottle*), and tool use (*Sweep Trash*). Across both in-distribution and out-of-distribution tasks, InternData-A1 consistently outperforms π -dataset by a 6.2% margin. In particular, consistent with superior performance on RoboTwin, the rich basic skill trajectories in InternData-A1 (e.g., pick, place, and move) promote performance on tasks composed mainly of these skills (e.g., *Place Markpen* and *Pass Bottle*). These results send an encouraging message: although simulations inevitably differ from real-world dynamics, many manipulation tasks exhibit natural robustness, as robotic hardware tolerates minor control inaccuracies, VLA models learn abstract visuomotor representations, and many tasks allow for approximate contact strategies. Consequently, large-scale simulated data can still provide transferable inductive priors for real-world manipulation.

Evaluating over Dexterous Tasks. We further evaluate the pretrained policy on four long-horizon dexterous tasks—folding garments, sorting industrial parts, unscrewing bottle caps, and zipping bag, directly challenging the core strengths of the official π_0 model. To ensure fair evaluation, we introduce a new embodiment, ARX AC One, which is unseen in both InternData-A1 and π -dataset. Despite the presence of novel objects, skills, and embodiments, InternData-A1 achieves performance comparable to π -dataset, as shown in the right panel of fig. 5. We suppose that, beyond the comprehensive coverage of atomic skills in InternData-A1 as discussed earlier, the substantial proportion of long-horizon tasks further enables the VLA model to explore a broader action space and develop continuous control abilities, which can then be effectively transferred to novel tasks and embodiments.

5.2. Benchmarking against open-source datasets

We further compare InternData-A1 with previous open-source datasets, including two real-world datasets OXE (O’Neill et al., 2024) and Agibot World (Bu et al., 2025a), and one simulation dataset, RoboCasa (Nasiriany et al., 2024). Due to resource constraints, we pre-train the π_0 model exclusively on each dataset for 500k iterations and evaluate performance on 49 simulated tasks and 2 real-world tasks. As shown in section 5.1, InternData-A1 demonstrates clear advantages across both simulation and real domains. In regard to RoboCasa (Nasiriany et al., 2024), while the RoboCasa demonstrates competitive performance in simulation evaluations, trailing InternData-A1 by only

Dataset	Domain	49 Sim Tasks		2 Real Tasks	
		Easy	Hard	Sort Rubbish	Pass Bottle
OXE (O'Neill et al., 2024)	Real	32.5%	11.0%	40.0%	36.7%
Agibot World (Bu et al., 2025a)	Real	52.5%	12.0%	53.3%	56.7%
RoboCasa (Nasiriany et al., 2024)	Sim	50.0%	11.0%	23.3%	13.3%
InternData-A1	Sim	60.0%	26.5%	90.0%	60.0%

Table 3. Performance comparison of models pretrained on InternData-A1 against other open-source datasets.

10%, its performance significantly drops in real-robot evaluation. In these real-world evaluations, InternData-A1 achieves an average improvement of 57.7% over RoboCasa. We attribute this significant gain to InternData-A1’s highly photorealistic rendering and abundant data amounts. Together with the comparison against the closed-source π -dataset, these results further validate the effectiveness of InternData-A1 as a strong pre-training source for VLA models.

6. Data Analysis

In this section, we aim to address two widely discussed questions regarding large-scale simulation data: (1) How well do the tasks in InternData-A1 transfer from simulation to the real world? (2) Which components of InternData-A1 contribute most to effective pre-training?

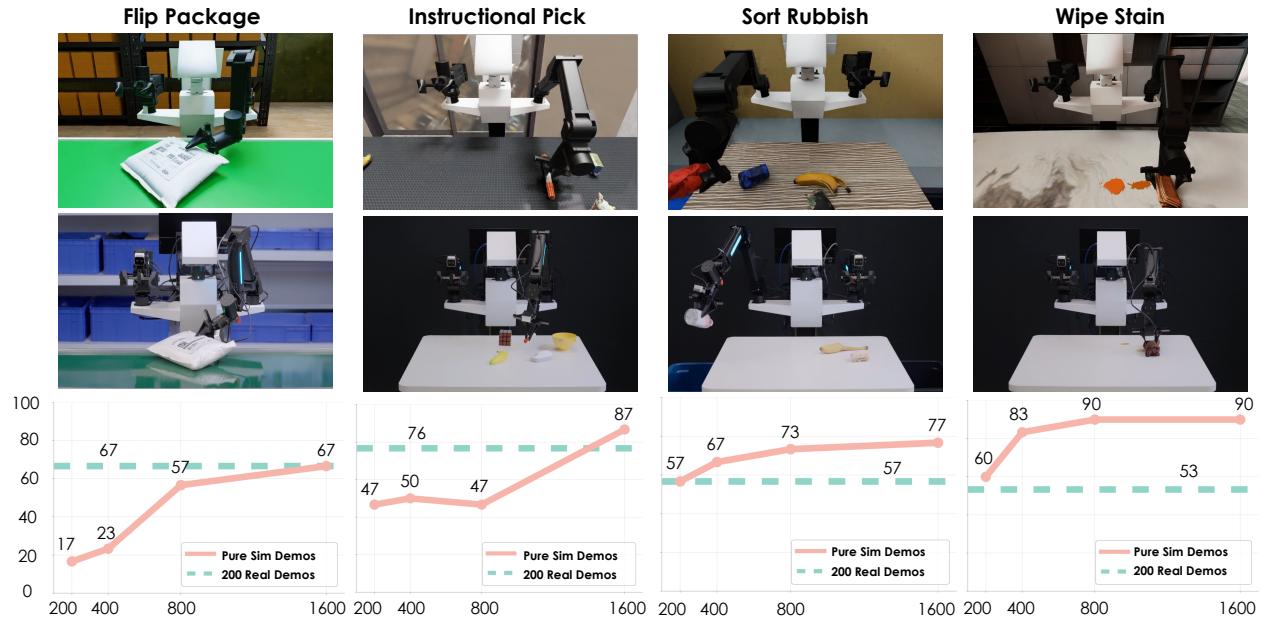


Figure 6. **Sim-to-real experimental setup.** Four tasks are evaluated via direct sim-to-real transfer, and simultaneously, the performance of training with simulation data is compared with that of training with real-world data.

Pure Simulation versus Pure Real. We investigate the zero-shot sim-to-real transfer ability on four representative tasks in InternData-A1. Starting from the same π_0 (InternData-A1) checkpoint, we post-train on 200–1,600 simulated episodes and 200 real episodes for each task, and evaluate each resulting policy over 30 rollouts. As shown in fig. 6, for regular tasks such as *Sort Rubbish* and

Wipe Stain, which mainly involve basic skills (pick, place, move), 200 simulated episodes already achieve performance comparable to 200 real ones. For more complex tasks such as *Flip Package* and *Instructional Pick*, which require dynamic-object manipulation and language grounding, around 1,600 simulated episodes are needed to match real-data performance. In other words, the simulation-to-real data ratio for equivalent performance narrows to within 8:1—and in some cases, even approaches 1:1. Although complex tasks demand more simulated data, our efficient synthesis pipeline achieves this at far lower time and economic costs than real-world collection. Moreover, we find that strong zero-shot sim-to-real performance does not require exact replicas of backgrounds, lighting, object textures, or table layouts. As shown in fig. 6, coarse alignment is sufficient, as long as camera views and joint action spaces are closely aligned. This robustness arises from InternData-A1’s photo-realistic rendering and extensive domain randomization, which narrow the visual gap and enable the policy to ignore irrelevant discrepancies between simulation and real environments.



Figure 7. **Additional sim-to-real performance.** Six tasks involving repeated pick-place, articulation, and bimanual coordination achieve over 50% success rates with only 500 simulated episodes.

Additional Sim-to-Real Tasks. To further examine the real-world potential of InternData-A1, we evaluate six additional tasks using 500 simulated episodes post-trained on π_0 (InternData-A1) and tested over 30 rollouts. As shown in fig. 7, the model achieves 50% success rates in pick-and-place tasks (*Make Sandwich*, *Pack*), 63% and 87% in articulation operations (*Close Box*, *Close Microwave*), and 60% and 57% in bimanual coordination tasks (*Sweep*, *Handover*). Together with the four tasks in fig. 6, ten representative tasks out of seventy achieve high sim-to-real success without any real data—a strong reflection of the data quality and minimal visual–physical gap in InternData-A1. Moreover, InternData-A1 is the first to demonstrate that, beyond single pick tasks as in GraspVLA (Deng et al., 2024), diverse and complex tasks can also be directly transferred via VLA from simulation to reality. These include multi-skill operations, articulation, bimanual manipulation, and even *Flip Package* in logistics scenes, which was previously realized using purely real data by the Helix (Cui et al., 2025) from the Figure Company. These findings highlight the strong potential of large-scale simulation data for VLA deployment in real world.

6.1. Data Component Ablation

What Matters in InternData-A1’s Composition. To uncover each component’s contribution to pre-training, we divide InternData-A1 into four parts: large-scale pick-and-place tasks (PnP, 30.61%), large-scale articulation manipulation (Art, 11.67%), base tasks involving fewer than three skills beyond simple pick-and-place (Base, 35.95%), and long-horizon tasks containing at least three skills (Long, 21.77%). Due to resource constraints, we pre-train π_0 for 0.5 epoch using both the full dataset and ablated versions (each with one component removed), and evaluate them on RoboTwin 2.0. As

Full	w.o. PnP.	w.o. Art.	w.o. Base.	w.o. Long.
Easy / Hard	Easy / Hard	Easy / Hard	Easy / Hard	Easy / Hard
58.0% / 25.0%	57.0% / 22.5%	55.5% / 19.5%	52.5% / 20.5%	54.0% / 19.0%

Table 4. **Data component ablation results.** “Pnp”, “Art”, “Base”, and “Long” represent pick-and-place tasks, articulation manipulation tasks, base tasks, and long-horizon task, respectively.

shown in section 6.1, the full-data model achieves the highest success rates, while removing any component causes a clear performance drop, indicating all four components contribute meaningfully.

Two observations stand out. First, although PnP (30.61%) and Base (35.95%) dominate the dataset, removing Base or Long tasks leads to a more significant decline than removing PnP. This suggests that for VLA pre-training, datasets composed solely of pick-and-place tasks (Chen et al., 2025b; Deng et al., 2024) are insufficient; maintaining task diversity, i.e., multi-skill compositions, is more beneficial for VLA training. Second, excluding articulation tasks, despite their smaller scale and limited object diversity, also yields a larger drop than excluding PnP. We suspect that articulated manipulation reaches a more diverse action space, as they are often influenced by diverse joint geometries and distorted arm configurations (e.g., pushing a distant drawer). At a higher level, combining these two findings, we hypothesize that **trajectory diversity** may serve as the core drive of effective pre-training. We leave a rigorous investigation for future research.

7. Conclusion and Limitation

This paper introduces InternData-A1, a large-scale, high-fidelity synthetic dataset for robotic manipulation. Using this dataset, we demonstrate for the first time that pre-training a Vision-Language-Action (VLA) model exclusively on synthetic data can match the performance of the leading VLA model, revealing the substantial potential of large-scale simulation. By open-sourcing the dataset and its generation pipeline, we aim to lower the barrier to large-scale robotic data access for the embodied AI community.

Limitation. Due to the limitations of physics simulators, it is challenging to simulate highly dexterous tasks, such as tying shoelaces or threading a needle. Future work will expand the dataset’s task diversity and dexterity, aiming to further establish large-scale simulation data as a cornerstone for the advancement of VLA models.

8. Acknowledgement

We acknowledge Hui Wang, Xing Shen, Baole Fang, and Yunsong Zhou for their efforts in the simulation and synthesis of deformable object manipulation. We thank Zeyu He, Yuanzhen Zhou, and Hengjie Li for their contributions in optimization and acceleration of data synthesis pipeline. We are grateful to Jiafei Cao for supporting the real-world experimental infrastructure, and to Liang Pan, Runyi Yu, Yuxi Wei, and Jingbo Wang for their pioneering work on early-stage development of the data synthesis pipeline. Finally, we acknowledge Lining Xu for her exceptional work in drafting and refining the manuscript.

References

- L. Beyer, A. Steiner, A. S. Pinto, A. Kolesnikov, X. Wang, D. Salz, M. Neumann, I. Alabdulmohsin, M. Tschannen, E. Bugliarello, et al. Paligemma: A versatile 3b vlm for transfer. In [ArXiv](#), 2024.
- J. Bjorck, F. Castañeda, N. Cherniadev, X. Da, R. Ding, L. Fan, Y. Fang, D. Fox, F. Hu, S. Huang, et al. Gr00t n1: An open foundation model for generalist humanoid robots. In [ArXiv](#), 2025.
- K. Black, N. Brown, D. Driess, A. Esmail, M. Equi, C. Finn, N. Fusai, L. Groom, K. Hausman, B. Ichter, et al. π_0 : A vision-language-action flow model for general robot control. In [RSS](#), 2024.
- A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, J. Dabis, C. Finn, K. Gopalakrishnan, K. Hausman, A. Herzog, J. Hsu, et al. Rt-1: Robotics transformer for real-world control at scale. In [ArXiv](#), 2022.
- Q. Bu, J. Cai, L. Chen, X. Cui, Y. Ding, S. Feng, S. Gao, X. He, X. Hu, X. Huang, et al. Agibot world colosseo: A large-scale manipulation platform for scalable and intelligent embodied systems. In [IROS](#), 2025a.
- Q. Bu, Y. Yang, J. Cai, S. Gao, G. Ren, M. Yao, P. Luo, and H. Li. Univla: Learning to act anywhere with task-centric latent actions. In [RSS](#), 2025b.
- C. Cheang, S. Chen, Z. Cui, Y. Hu, L. Huang, T. Kong, H. Li, Y. Li, Y. Liu, X. Ma, et al. Gr-3 technical report. In [ArXiv](#), 2025.
- C.-L. Cheang, G. Chen, Y. Jing, T. Kong, H. Li, Y. Li, Y. Liu, H. Wu, J. Xu, Y. Yang, et al. Gr-2: A generative video-language-action model with web-scale knowledge for robot manipulation. In [ArXiv](#), 2024.
- A. H. Chen, Z. Liu, Y. Yang, and C. Yuksel. Vertex block descent. In [TOG](#), 2024.
- T. Chen, Z. Chen, B. Chen, Z. Cai, Y. Liu, Z. Li, Q. Liang, X. Lin, Y. Ge, Z. Gu, et al. Robotwin 2.0: A scalable data generator and benchmark with strong domain randomization for robust bimanual robotic manipulation. In [ArXiv](#), 2025a.
- X. Chen, Y. Chen, Y. Fu, N. Gao, J. Jia, W. Jin, H. Li, Y. Mu, J. Pang, Y. Qiao, et al. Internvla-m1: A spatially guided vision-language-action framework for generalist robot policy. In [ArXiv](#), 2025b.
- C. Cui, P. Ding, W. Song, S. Bai, X. Tong, Z. Ge, R. Suo, W. Zhou, Y. Liu, B. Jia, et al. Openhelix: A short survey, empirical analysis, and open-source dual-system vla model for robotic manipulation. In [ArXiv](#), 2025.
- M. Deitke, D. Schwenk, J. Salvador, L. Weihs, O. Michel, E. VanderBilt, L. Schmidt, K. Ehsani, A. Kembhavi, and A. Farhadi. Objaverse: A universe of annotated 3d objects. In [CVPR](#), 2023.
- S. Deng, M. Yan, S. Wei, H. Ma, Y. Yang, J. Chen, Z. Zhang, T. Yang, X. Zhang, W. Zhang, et al. Graspvla: a grasping foundation model pre-trained on billion-scale synthetic action data. In [ArXiv](#), 2024.
- H.-S. Fang, C. Wang, H. Fang, M. Gou, J. Liu, H. Yan, W. Liu, Y. Xie, and C. Lu. Anygrasp: Robust and efficient grasp perception in spatial and temporal domains. In [TRO](#), 2023.
- H. Geng, H. Xu, C. Zhao, C. Xu, L. Yi, S. Huang, and H. Wang. Gapartnet: Cross-category domain-generalizable object perception and manipulation via generalizable and actionable parts. In [CVPR](#), 2023.

- J. Gu, F. Xiang, X. Li, Z. Ling, X. Liu, T. Mu, Y. Tang, S. Tao, X. Wei, Y. Yao, et al. Maniskill2: A unified benchmark for generalizable manipulation skills. In [ArXiv](#), 2023.
- P. Hua, M. Liu, A. Macaluso, Y. Lin, W. Zhang, H. Xu, and L. Wang. Gensim2: Scaling robot data generation with multi-modal and reasoning llms. In [ArXiv](#), 2024.
- P. Intelligence, K. Black, N. Brown, J. Darpinian, K. Dhabalia, D. Driess, A. Esmail, M. Equi, C. Finn, N. Fusai, et al. $\pi_{0.5}$: a vision-language-action model with open-world generalization. In [CORL](#), 2025.
- S. James, Z. Ma, D. R. Arrojo, and A. J. Davison. Rlbench: The robot learning benchmark & learning environment. In [RAL](#), 2020.
- T. Jiang, T. Yuan, Y. Liu, C. Lu, J. Cui, X. Liu, S. Cheng, J. Gao, H. Xu, and H. Zhao. Galaxea open-world dataset and g0 dual-system vla model. In [ArXiv](#), 2025.
- Z. Jin, Z. Che, Z. Zhao, K. Wu, Y. Zhang, Y. Zhao, Z. Liu, Q. Zhang, X. Ju, J. Tian, et al. Artvip: Articulated digital assets of visual realism, modular interaction, and physical fidelity for robot learning. In [ArXiv](#), 2025.
- A. Khazatsky, K. Pertsch, S. Nair, A. Balakrishna, S. Dasari, S. Karamcheti, S. Nasiriany, M. K. Srirama, L. Y. Chen, K. Ellis, et al. Droid: A large-scale in-the-wild robot manipulation dataset. In [RSS](#), 2024.
- 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. In [CORL](#), 2024.
- H. Li, S. Yang, Y. Chen, Y. Tian, X. Yang, X. Chen, H. Wang, T. Wang, F. Zhao, D. Lin, et al. Cronusvla: Transferring latent motion across time for multi-frame prediction in manipulation. In [AAAI](#), 2025.
- X. Lian, Z. Yu, R. Liang, Y. Wang, L. R. Luo, K. Chen, Y. Zhou, Q. Tang, X. Xu, Z. Lyu, et al. Infinite mobility: Scalable high-fidelity synthesis of articulated objects via procedural generation. In [ArXiv](#), 2025.
- B. Liu, Y. Zhu, C. Gao, Y. Feng, Q. Liu, Y. Zhu, and P. Stone. Libero: Benchmarking knowledge transfer for lifelong robot learning. In [NIPS](#), 2023.
- A. Mandlekar, S. Nasiriany, B. Wen, I. Akinola, Y. Narang, L. Fan, Y. Zhu, and D. Fox. Mimicgen: A data generation system for scalable robot learning using human demonstrations. In [ArXiv](#), 2023.
- O. Mees, L. Hermann, E. Rosete-Beas, and W. Burgard. Calvin: A benchmark for language-conditioned policy learning for long-horizon robot manipulation tasks. In [RAL](#), 2022.
- S. Nasiriany, A. Maddukuri, L. Zhang, A. Parikh, A. Lo, A. Joshi, A. Mandlekar, and Y. Zhu. Robocasa: Large-scale simulation of everyday tasks for generalist robots. In [CVPR](#), 2024.
- A. O'Neill, A. Rehman, A. Maddukuri, A. Gupta, A. Padalkar, A. Lee, A. Pooley, A. Gupta, A. Mandlekar, A. Jain, et al. Open x-embodiment: Robotic learning datasets and rt-x models: Open x-embodiment collaboration 0. In [ICRA](#), 2024.
- D. Qu, H. Song, Q. Chen, Z. Chen, X. Gao, X. Ye, Q. Lv, M. Shi, G. Ren, C. Ruan, et al. Eo-1: Interleaved vision-text-action pretraining for general robot control. In [ArXiv](#), 2025.
- B. Sundaralingam, S. K. S. Hari, A. Fishman, C. Garrett, K. V. Wyk, V. Blukis, A. Millane, H. Oleynikova, A. Handa, F. Ramos, N. Ratliff, and D. Fox. curobo: Parallelized collision-free minimum-jerk robot motion generation, 2023.

- Y. Tian, S. Yang, J. Zeng, P. Wang, D. Lin, H. Dong, and J. Pang. Predictive inverse dynamics models are scalable learners for robotic manipulation. In ICLR, 2025.
- H. R. Walke, K. Black, T. Z. Zhao, Q. Vuong, C. Zheng, P. Hansen-Estruch, A. W. He, V. Myers, M. J. Kim, M. Du, et al. Bridgedata v2: A dataset for robot learning at scale. In CORL, 2023.
- H. Wang, J. Chen, W. Huang, Q. Ben, T. Wang, B. Mi, T. Huang, S. Zhao, Y. Chen, S. Yang, et al. Grutopia: Dream general robots in a city at scale. In ArXiv, 2024.
- K. Wu, C. Hou, J. Liu, Z. Che, X. Ju, Z. Yang, M. Li, Y. Zhao, Z. Xu, G. Yang, et al. Robomind: Benchmark on multi-embodiment intelligence normative data for robot manipulation. In RSS, 2024.
- T. Wu, J. Zhang, X. Fu, Y. Wang, J. Ren, L. Pan, W. Wu, L. Yang, J. Wang, C. Qian, et al. Omniobject3d: Large-vocabulary 3d object dataset for realistic perception, reconstruction and generation. In CVPR, 2023.
- S. Yang, H. Li, Y. Chen, B. Wang, Y. Tian, T. Wang, H. Wang, F. Zhao, Y. Liao, and J. Pang. Instructvla: Vision-language-action instruction tuning from understanding to manipulation. In ArXiv, 2025a.
- Y. Yang, Z. Cai, Y. Tian, J. Zeng, and J. Pang. Gripper keypose and object pointflow as interfaces for bimanual robotic manipulation. In RSS, 2025b.
- A. Zhai, B. Liu, B. Fang, C. Cai, E. Ma, E. Yin, H. Wang, H. Zhou, J. Wang, L. Shi, et al. Igniting vlms toward the embodied space. In ArXiv, 2025.
- J. Zheng, J. Li, Z. Wang, D. Liu, X. Kang, Y. Feng, Y. Zheng, J. Zou, Y. Chen, J. Zeng, et al. X-vla: Soft-prompted transformer as scalable cross-embodiment vision-language-action model. In ArXiv, 2025.
- B. Zitkovich, T. Yu, S. Xu, P. Xu, T. Xiao, F. Xia, J. Wu, P. Wohlhart, S. Welker, A. Wahid, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. In CORL, 2023.

Table 5. Task Statistics Across Robots.

Task Name	Franka	ARX Lift-2	Agilex Split Aloha	Genie-1	Sum
Articulation Tasks (11.67%)					74,415
Close The Electric Cooker	1776				
Close The Laptop	578				
Close The Pot	2595				
Close The Trashcan	2996				
Close The Microwave	2496	6831	4367		
Open The Laptop	416				
Open The Pot	3250				
Open The Trashcan	3507				
Open The Microwave	2139	5148	4817		
Pull The Storage Furniture		4368	4776		
Push The Storage Furniture		4548	4775		
Rotate The Hearth		6046	4653		
Open Microwave From Scratch			1501		
Heat Food In Microwave		108			
Close The Package	2724				
Long-horizon Tasks (21.77%)					138,782
Clean Dirt With Brown Cloth		3000	3000		
Clean Dirt With Sponge		3000	3000		
Clean Dirt With White Cloth		3000	3000		
Collect Three Glues		2000	2000		
Gather Three Teaboxes		110	2000		
Handover Objects		7863			
Pack In Objects		4052	2345		
Pack Out Objects			2070		
Sort The Rubbish		4579	8860		
Stack Multiple Objects		4867	4664		
Sweep The Trash	2344	528	1626		
Put Trash In Trashcan	1480				
Collaborate Assemble Beef Sandwich		4854			
Stack A Beef Sandwich		4271	670		
Store Objects In Drawer		2822			
Collaborate Assemble Ham Sandwich		3168			

Continued on next page

Task Name	Franka	ARX Lift-2	Agilex Split Aloha	Genie-1	Sum
Continues Pick And Place	20036	15000	18573		
Base Tasks (35.95%)					229,168
Track The Target	2959	2954	3000		
Organize Three Brushes	5064	2000	2000		
Organize Alarm Clocks	5111	2000	2000		
Organize Colorful Cups	5097	2000	2000		
Organize Three Glues	5120				
Collect Shoes	5114	2000	2000		
Organize Three Teaboxes	5119				
Sort Table Waste	5117	214	2000		
Store Eggs				4244	
Take Shelf Items To Cart				6040	
Pick Beef Sandwich On Conveyor		6658	6647		
Pick Ham Sandwich On Conveyor		4092	4220		
Fold Long Shirts		731			
Fold Short Shirts		492			
Fold Towels		500			
Fold Short Pants		750			
Flip Package On Conveyor		4806			
Pick Package On Conveyor		4900			
Hang Cups On Rack		5000	5000		
Insert Flower In Vase		5000	4986		
Insert Markpen In Pen-holder		5000	5000		
Pour Baijiu		4999	4999		
Pour Redwine		5000	5000		
Pour Water		5000	5000		
Pick The Priced Item	5105	2000	2000		
Select A Drink	5121	2000	2000		
Stack Two Boxes		2270	2429		
Sort Tray On Rack		3851	3444		
Store Toothbrushes		1396			
Arrange The Tableware		650			
Recovery Pick Objects		10969			
Watering Plants		5000	5000		
Scan The QRcode				4000	
Sort Metallic Objects		2500	2500		

Continued on next page

Task Name	Franka	ARX Lift-2	Agilex Split Aloha	Genie-1	Sum
Pick and Place Tasks (30.61%)					195,133
Single Arm Pick	24598	38865	39219	21695	
Parallel Pick And Place		15687	18497	10381	
Grasp Functional Part				4833	
Multiple Pick And Place	21358				
Overall Trajectories					637,498
Overall Frames					401,430,981
Overall Hours					7433.91

A. Detailed Data Statistics

We report the complete dataset statistics in table 5. In total, the dataset contains 4 embodiments, 70 tasks, 637,498 trajectories, 401,430,981 frames, and 7,433.91 hours of interaction data. As outlined above, the dataset is organized into four categories: **Articulation**, **Long-horizon**, **Base**, and **Pick and Place**. These categories comprise 74,415, 138,782, 229,168, and 195,133 trajectories, accounting for 11.67%, 21.77%, 35.95%, and 30.61% of the dataset, respectively. For each task, we report the exact number of trajectories contributed by each embodiment. See table 5 for detailed per-task and per-embodiment statistics.

B. Detailed Data Synthesis

We present an example task config below. Following the exact task config, we elaborate on each part in our data synthesis in detail.

B.1. Environment Construction

As shown in the configuration, we set the room environment using `defaults/arenas@arena` and select a dining-room layout in the outer script. We load the Agilex Split Aloha robot—one of our four embodiments—and specify its motion planner via the `robot_file`. We then retrieve two task-relevant assets, the plate and plate shelf, from our asset library, where each asset is automatically annotated with gravity parameters, collision properties, and grasp poses. After obtaining the two objects, we assign their initial translations and orientations. Objects belonging to the same category share a unified canonical pose definition.

B.2. Skill Composition

As shown in the configuration—particularly in the `skills/split_aloha` section—we construct tasks by composing skills either sequentially or in parallel. Users can simply copy and paste different skill blocks to assemble a task. For example, a complete task may be formed by chaining together `pick`, `goto_pose`, `pick`, `gripper_action` (close or open), `home`, and `place`. The framework supports both sequential execution and parallel execution (e.g., one gripper opens while the other closes), enabling users to specify diverse task requirements. Users may also define task-level constraints; for instance, in a placement operation, we enforce `align_pick_obj_axis` and `align_place_obj_axis` to be parallel to ensure accurate insertion. Similarly, `x_ratio_range`

and `y_ratio_range` can be used to specify the target insertion layer. All script-level policies have undergone substantial refinement. For manual tuning, users may configure grasp-pose filtering rules (`filter_x_dir`, `filter_y_dir`, `filter_z_dir`) and adjust parameters such as `post_grasp_offset_min`, and `place_z_offset` to ensure stable grasping and placement while avoiding unsafe motions.

B.3. Domain Randomization

For visual domain randomization, we provide options in `env_map`, allowing light intensity and rotation to be perturbed within predefined ranges. We also support camera extrinsic randomization, where camera poses are perturbed by up to 5° in rotation and 5 cm in translation. Room scenes can be randomized by sampling from the specified room types. For objects, replacements can be sampled from assets within the same category. At the trajectory level, we define a spatial region in which target objects and robots are initialized with randomized poses for each episode. Additionally, the `robots` configuration allows specifying the mean and standard deviation of the home configuration, enabling diverse initial joint states. Within the skill definitions, we further introduce loose filtering ranges for grasp and placement poses. All poses that satisfy these constraints are retained, and a final pose is selected randomly. Together, these mechanisms significantly enhance trajectory diversity within each task.

C. Policy Training Details

During real-world training, we pretrain a new π_0 model, initialized with Paligemma weights and a scratched action expert, on InternData-A1 for 680k iterations using 32 A100 GPUs (closely matches the 700k iteration steps of the official π_0 checkpoint trained on the π -dataset). For 10 sim-to-real experiments and 9 real-world tasks, we start from the 680k π_0 (InternData-A1) checkpoint and perform post-training for 30k iterations on 8 GPUs for regular tasks and sim-to-real tasks. For dexterous tasks, we trained for 100k iterations. Key training hyperparameters are summarized in table 6.

Hyperparameters	Pre-training	Fine-tuning
Batch Size(Total)	512	128
Learning Rate	5e-5	2.5e-5
Learning Rate Schedule	Constant	Cosine Decay
Training Steps	680k	30k(Regular)/100k(Dexterous)

Table 6. Training hyperparameters.

D. Real-World Experiments

In this section, we describe the real-world and sim-to-real tasks in detail. For both experiments, we post-train a JAX-version π_0 (InternData-A1) model for 30k iterations and use the 30k checkpoint for evaluation. For each task, we define 15 evaluation settings, and to reduce stochasticity, we run two trials per setting. In total, each task is evaluated with 30 rollouts, and we report the average success rate.

D.1. Real Task Description

Place Markpen. The Genie-1 robot is required to pick a black marker with its right arm and place it into a pen holder. This task evaluates the model’s fundamental pick-and-place capabilities. A trial is considered successful only if the marker is placed precisely and fully inside the pen holder.

Pass Bottle. The Genie-1 robot is required to pick up a black tea bottle, lift it upright, and hand it to a nearby person with the right arm. The robot may release its gripper only when the human presents their hand. This task evaluates the model’s fundamental abilities in picking, lifting, and human–robot interaction. A trial is considered successful only if the bottle is successfully transferred to the human and the robot releases its gripper accordingly.

Heat Sandwich. The ARX Lift-2 robot must open the oven with its left arm, pick up the plate containing the sandwich, place it into the oven using its right arm, and then close the oven with its left arm. This task assesses the model’s ability to operate articulated objects. A trial is considered successful only if the plate is correctly inserted into the oven and the oven door is fully closed.

Sort Rubbish. The ARX Lift-2 robot must use its right arm to place all recyclable waste into the right bin and all non-recyclable waste into the left bin. This task evaluates the model’s ability to handle diverse object layouts and perform repetitive pick-and-place operations. A trial is considered successful only if all waste items are fully and correctly sorted.

Sweep Trash. The ARX Lift-2 robot must grasp the dustpan with its right arm and the broom with its left arm. It then uses the broom to sweep all crumpled paper balls into the dustpan. Afterwards, the robot empties the dustpan into the left rubbish bin. Finally, it releases both grippers and returns to the home position. A trial is considered successful only if every step is finished successfully.

Sort Parts. The ARX Lift-2 robot must sort four types of small industrial components into four designated containers. These components include small nuts, assembly parts, and small screws. Each arm is responsible for sorting two categories. A trial is considered successful only if all components are placed into their correct containers.

Unscrew Cap The ARX AC One robot must grasp the tea bottle with its left arm and move it to the designated middle zone. It then uses its right arm to approach the bottle cap and unscrew it. A trial is considered successful only if the cap is fully removed.

Fold Cloths. The ARX AC One robot must fold the cloth into its designated final shape with both hands. A trial is considered successful only if the cloth is folded correctly. **Zip Bag.** The ARX AC One robot must use its left arm to open the bag, place all designated objects inside, and then zip it up. A trial is considered successful only if the bag is fully and correctly zipped.

D.2. Sim-to-real Task Description

Flip Package. A package is placed on the conveyor and moves toward the robot. The ARX Lift-2 robot must grasp the package with its right arm, flip it over, and place it back onto the conveyor. It must then grasp the package with its left arm and scan the QR code using the robot-mounted camera. A trial is considered successful only if all steps are completed correctly.

Instructional Pick. Eight types of objects are placed on the table. A trial is considered successful only if the robot correctly picks the target object specified by the command. **Sort Rubbish.** This is the same task as described before.

Wipe Stain. The ARX Lift-2 robot uses its left arm to pick up the towel and wipe stains located in one or two clusters. A trial is considered successful only if all stains are completely removed.

Sandwich. The ARX Lift-2 robot uses its right arm to grasp a piece of bread and place it on the plate. It then uses its left arm to grasp a piece of beef and place it on the bread, followed by using the right arm again to place another piece of bread on top of the beef. A trial is considered successful only if the sandwich is assembled correctly and neatly.

Box. The ARX Lift-2 robot sequentially closes the box lids with its right and left arms. A trial is considered successful upon complete closure.

Microwave. The ARX Lift-2 robot uses the right arm to close the microwave lid. A trial is considered successful upon complete closure.

Pack. The ARX Lift-2 robot manipulates objects and places them into a box using its right and left arms. A trial is considered successful only after all objects have been placed inside.

Sweep. This is the same task as described before.

Handover. The ARX Lift-2 robot uses its left arm to pick up a long-shaped object and hands it over to the right arm, which then places it into the box. A trial is considered successful only upon the object's transfer into the box.

```

1 defaults:
2   - _self_
3   - world
4   - logger
5   - ../arenas@arena: scene_arena
6   - ../cameras@astra: astra
7   - ../cameras@realsense_d455_v3: realsense_d455_v3
8
9 name: banana_base_task
10 asset_root: assets
11 task: BananaBaseTask
12 task_id: 0
13
14 offset: null
15 render: True
16
17 env_map:
18   envmap_lib: envmap_lib
19   apply_randomization: True
20   intensity_range: [4000, 7000]
21   rotation_range: [0, 180]
22
23 robots:
24   -
25     name: "split_aloha"
26     target_class: SplitAloha
27     path: "split_aloha_mid_360/robot_task13.usd"
28     euler: [0.0, 0.0, 90.0]
29     robot_file:
30       - curobo/src/curobo/content/configs/robot/piper100_left_arm.yml
31       - curobo/src/curobo/content/configs/robot/piper100_right_arm.yml
32     left_joint_home: [0.00484993, 0.34198609, -0.14007858, 0.01680429,
33       0.14391101, -0.00252178]
34     right_joint_home: [0.00484993, 0.34198609, -0.14007858, 0.01680429,
35       0.14391101, -0.00252178]
36     left_joint_home_std: [0.12513939, 0.24539099, 0.24468172, 0.23398885,
37       0.2710117, 0.21726329]
38     right_joint_home_std: [0.12513939, 0.24539099, 0.24468172, 0.23398885,
39       0.2710117, 0.21726329]
40
41 objects:
42   -
43     name: arcode_plate_blue
44     path: assets/plate/plate_blue/Aligned_obj.usd
45     target_class: RigidObject
46     dataset: arcode
47     category: plate
48     prim_path_child: Aligned
49     translation: [0.0, 0.0, 0.0]
50     euler: [90.0, 0.0, 0.0]
51     scale: [1.0, 1.0, 1.0]
52     apply_randomization: True
53   -
54     name: arcode_plate_shelf
55     path: assets/plate_shelf/shelf_0/Aligned_obj.usd
56     target_class: RigidObject

```

```

57     euler: [90.0, 0.0, 0.0]
58     scale: [1.0, 1.0, 1.0]
59     apply_randomization: False
60
61 regions:
62   -
63     object: ${robots.0.name}
64     target: table
65     random_type: A_on_B_region_sampler
66     random_config:
67       pos_range: [
68         [0.0, -0.86, -0.765],
69         [0.0, -0.86, -0.765]
70       ]
71       yaw_rotation: [0.0, 0.0]
72   -
73     object: arcode_plate_blue
74     target: table
75     random_type: A_on_B_region_sampler
76     random_config:
77       pos_range: [
78         [0.125, -0.20, 0.005],
79         [0.25, -0.10, 0.005]
80       ]
81       yaw_rotation: [0, 0]
82   -
83     object: arcode_plate_shelf
84     target: table
85     random_type: A_on_B_region_sampler
86     random_config:
87       pos_range: [
88         [-0.25, -0.20, 0.005],
89         [-0.15, -0.10, 0.005]
90       ]
91       yaw_rotation: [0, 0]
92
93 cameras:
94   -
95     name: ${robots.0.name}_hand_left
96     translation: [0.0, 0.08, 0.05]
97     orientation: [0.0, 0.0, 0.965, 0.259]
98     camera_axes: usd
99     params: ${astral}
100    parent: "${robots.0.name}/split_aloha_mid_360_with_piper/fl/link6"
101    apply_randomization: True
102    max_translation_noise: 0.03
103    max_orientation_noise: 5.0
104
105   -
106     name: ${robots.0.name}_hand_right
107     translation: [0.0, 0.08, 0.04]
108     orientation: [0.0, 0.0, 0.972, 0.233]
109     camera_axes: usd
110     params: ${astral}
111     parent: "${robots.0.name}/split_aloha_mid_360_with_piper/fr/link6"
112     apply_randomization: True
113     max_translation_noise: 0.03
114     max_orientation_noise: 5.0
115
116   -
117     name: ${robots.0.name}_head

```

```

118     translation: [0.0, -0.00818, 0.1]
119     orientation: [0.658, 0.259, -0.282, -0.648]
120     camera_axes: usd
121     params: ${realsense_d455_v3}
122     parent:
123         "${robots.0.name}/split_aloha_mid_360_with_piper/top_camera_link"
124     apply_randomization: True
125     max_translation_noise: 0.03
126     max_orientation_noise: 5.0
127
128 data:
129     save_root_path: "InternData-A1/sim/raw_data"
130     task_dir: "Sort Tray On Rack"
131     language_instruction: "Pick the plate, make the handover and place it on
132         the water cooling holder"
133     detailed_language_instruction: "Pick the plate with the right arm, make
134         the handover to the left arm, and then place it on the water cooling
135         holder."
136     collect_info: ""
137     version: "v3.0, head camereea 1280x720, wrist 640x480, y 45 degrees"
138     update: True
139     max_episode_length: 4000
140
141 skills:
142     -
143         split_aloha:
144             -
145                 right:
146                     -
147                         name: pick
148                         objects: [arcade_plate_blue]
149                         filter_x_dir: ["upward", 90, 45]
150                         filter_y_dir: ["forward", 40]
151                         filter_z_dir: ["downward", 110, 140]
152                         t_eps: 0.01
153                         o_eps: 1
154                         close_wait_steps: 10
155                         post_grasp_offset_min: 0.1
156                         post_grasp_offset_max: 0.1
157                         direction_to_obj: right
158
159             -
160                 name: goto_pose
161                 frame: robot
162                 gripper_action: close_gripper
163                 translation: [0.3, 0.13, 0.15]
164                 quaternion: [-0.15, -0.37, -0.84, -0.36]
165
166         -
167             left:
168                 -
169                     name: pick
170                     objects: [arcade_plate_blue]
171                     filter_y_dir: ["upward", 40]
172                     filter_z_dir: ["forward", 90, 45]
173                     close_wait_steps: 10
174                     t_eps: 0.01
175                     o_eps: 1
176                     post_grasp_offset_min: 0.0
177                     post_grasp_offset_max: 0.0
178                     direction_to_obj: left

```

```

175
176
177     - left:
178         - name: gripper__action
179             action_type: close
180     right:
181         - name: gripper__action
182             action_type: open
183
184     -
185     right:
186         - name: home
187
188     -
189     left:
190         -
191             name: place
192             place_direction: vertical
193             objects: [arcade_plate_blue, arcade_plate_shelf]
194             filter_y_dir: ["upward", 60, 0]
195             filter_z_dir: ["forward", 90, 30]
196             position_constraint: object
197             x_ratio_range: [0.5, 0.5]
198             y_ratio_range: [0.8, 0.8]
199             align_pick_obj_axis: [0, 1, 0]
200             align_place_obj_axis: [0, 0, 1]
201             align_obj_tol: 10
202             pre_place_z_offset: 0.15
203             place_z_offset: 0.01

```

Listing 1. A Task Config Example on Sort Tray On Rack.