Rethinking Visual-Language-Action Model Scaling: Alignment, Mixture, and Regularization

Ye Wang[12*] Sipeng Zheng[2*] Hao Luo[23*] Wanpeng Zhang[23*] Haoqi Yuan[23] Chaoyi Xu[2]
Haiweng Xu[23] Yicheng Feng[23] Mingyang Yu[1] Zhiyu Kang[1] Zongqing Lu[23] Qin Jin[1†]
[1]Renmin University of China [2]BeingBeyond [3]Peking University
[*]Equal contribution [†]Corresponding author

***Abstract*—While Vision–Language–Action (VLA) models show strong promise for generalist robot control, it remains unclear whether—and under what conditions—the standard "scale data" recipe translates to robotics, where training data is inherently heterogeneous across embodiments, sensors, and action spaces. We present a systematic, controlled study of VLA scaling that revisits core training choices for pretraining across diverse robots. Using a representative VLA framework that combines a vision–language backbone with flow-matching, we ablate key design decisions under matched conditions and evaluate in extensive simulation and real-robot experiments. To improve the reliability of real-world results, we introduce a Grouped Blind Ensemble protocol that blinds operators to model identity and separates policy execution from outcome judgment, reducing experimenter bias. Our analysis targets three dimensions of VLA scaling. (1) Physical alignment: we show that a unified end-effector (EEF)-relative action representation is critical for robust cross-embodiment transfer. (2) Embodiment mixture: we find that naively pooling heterogeneous robot datasets often induces negative transfer rather than gains, underscoring the fragility of indiscriminate data scaling. (3) Training regularization: we observe that intuitive strategies, such as sensory dropout and multi-stage fine-tuning, do not consistently improve performance at scale. Together, this study challenge some common assumptions about embodied scaling and provide practical guidance for training large-scale VLA policies from diverse robotic data. Website: https://research.beingbeyond.com/rethink_vla**

I. Introduction

Vision–Language–Action (VLA) models [4, 16] have become a promising direction for general-purpose embodied AI. Following the scaling trends of vision–language models (VLMs) [8, 1], robotics is moving from single-task policies to generalist policies that aim to solve many tasks across different environments. This shift is supported by recent large robotic datasets [30, 5] with thousands of hours from diverse robot platforms, collected in both simulation and the real world.

A common belief is that, as in language modeling, improved generalization in robotics will primarily emerge from scaling data and model size. However, scaling in robotics introduces physical and system-level challenges that do not arise in text or image generation. Robotic data is inherently heterogeneous: robots differ in kinematics, joint limits, control frequencies, sensing modalities, and action spaces, which are often incompatible by design. As a result, it remains unclear whether scaling heterogeneous robotic data reliably leads to positive transfer, or whether embodiment differences introduce interference that limits performance.

These challenges raise several open questions that are central to scaling VLA models: *What action representations best align heterogeneous embodiments? When does mixing data across robots help, and when does it hurt?* And *how effective are commonly used regularization strategies when applied at scale in embodied settings?*

In this work, we address these questions through a controlled empirical study of VLA scaling. Instead of proposing a new architecture, we build a controlled testbed based on a representative VLA framework that combines a VLM backbone with flow matching [27, 3]. We treat the training pipeline itself as the primary object of study and systematically ablate key design choices under matched conditions. Our study is organized around three pillars: **(1) Physical Alignment.** We examine how coordinate frames and action representations affect general control performance across different setups. **(2) Embodiment Mixture.** We study how performance changes with different mixtures of heterogeneous data. Specifically, we investigate whether pooling real-world trajectories across different robots promotes positive transfer, or if cross-embodiment variation can introduce interference. **(3) Training Regularization.** We evaluate the scalability of training modifications, such as sensory dropout and curriculum learning, to determine if these approaches provide tangible benefits in a large-scale pretraining regime.

Reliable evaluation is also critical for drawing conclusions about scaling. Real-world testing can be influenced by experimenter bias, including operator familiarity with a policy and small adjustments during execution. To ensure reliable evaluation, we propose a Grouped Blind Ensemble protocol for real-world experiments: operators execute tasks without knowing which model variant is being tested, and we separate model inference from outcome judgment. By decoupling policy execution from outcome judgment and blinding operators to model identity, this protocol reduces human bias and improves the objectivity of real-robot evaluation.

Our main findings provide practical guidance for training large-scale VLA models: 1) End-effector (EEF) relative action space consistently outperforms other representations, establishing itself as a reliable default for VLA training; 2) Scaling cross-embodiment data is challenging because indiscriminately mixing heterogeneous robotic data frequently degrades performance, highlighting the fragility of naive data scaling; 3) Intuitive training modifications, such as randomized sensory

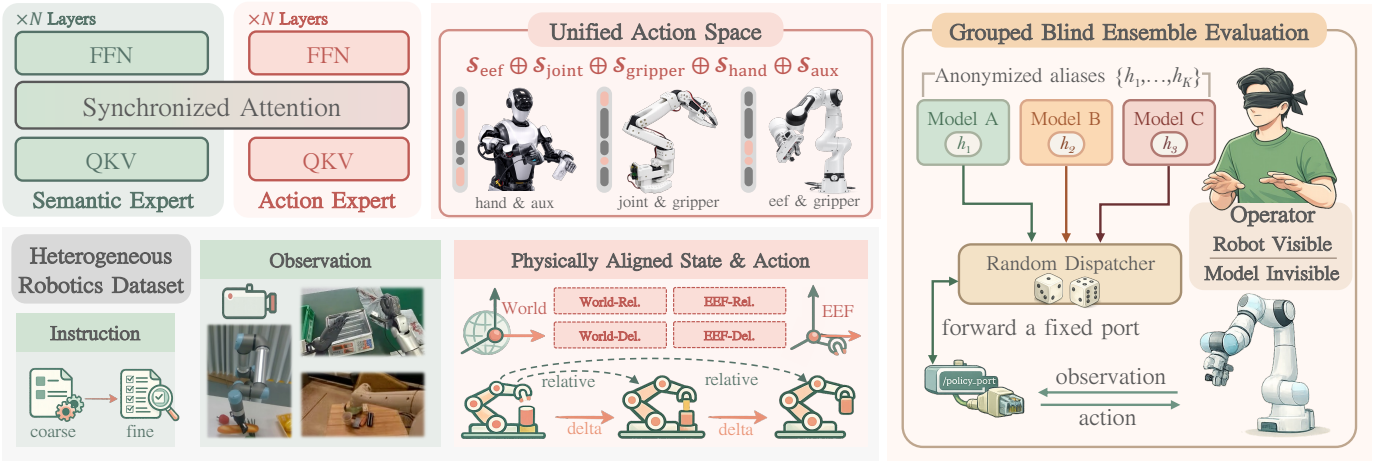

Fig. 1: Overview of our systematic VLA analysis framework, comprising the Mixture-of-Transformers architecture, physically aligned action spaces, and the Grouped Blind Ensemble protocol.

dropouts and multi-stage curricula, do not reliably translate into gains at scale; 4) Our proposed grouped blind ensemble protocol for real-world evaluation significantly reduces human bias in measuring robotic performance.

II. Related Work

Vision-Language-Action Models. Research in robotic manipulation has evolved significantly, transitioning from narrowly defined, single-task policies to more versatile systems that are trained on extensive and varied datasets [2, 9, 11, 32]. Vision–Language–Action (VLA) models [4, 44, 20, 17, 7] exemplify this trend by transferring the perceptual and reasoning priors of pre-trained vision-language models (VLMs) [34, 22, 37, 43, 42, 13, 12] into embodied control through fine-tuning on robot demonstrations. Although many VLAs share similar backbone architectures, their action generation choices vary widely. Early works often cast continuous control as discrete tokens and decode actions autoregressively to better match VLM-style objectives [20, 31]; this design eases transfer but can increase inference cost and struggle with high-precision, high-DoF execution. More recent approaches instead generate action chunks with diffusion-style objectives [21, 17, 26, 23], including π_0 [3] and GR00T-N1 [29], built upon denoising diffusion and related flow-matching formulations [14, 24]. Despite these advances, it remains challenging to consistently translate high-level VLM priors into fine-grained control across diverse embodiments. In contrast to proposing a new architecture, our work focuses on how training choices such as action parameterization, cross-embodiment data mixture, and regularization affect VLA scaling under controlled conditions.

Robotic Data for Pre-Training. The efficacy of Vision-Language-Action (VLA) training is closely linked to the scale and diversity of the robotic data utilized. Numerous initiatives, such as Open X-Embodiment [30], Droid [19] and BridgeData [36], compile significant demonstration hours across various tasks, environments, and platforms. Community-driven datasets and benchmarking efforts further enhance accessibility and coverage in this domain [33]. To better capture fine-grained manipulation, recent datasets are focusing on bimanual coordination, dexterous actions, and more sophisticated interaction scenarios. Notable examples include AgiBot World [5], and RoboCOIN [40] and RoboMIND [39, 15], while Open Galaxea [18] is dedicated to mobile manipulation tasks. Nevertheless, much teleoperated data is still collected in constrained settings. Synthetic data [41, 6] can alleviate data scarcity, but the sim-to-real gap remains a persistent obstacle. More fundamentally, these datasets are heterogeneous by construction, spanning different kinematic structures, sensing/control conventions, and action spaces, which complicates naive pooling and transfer. Our study directly examines this heterogeneity and provides controlled evidence on when mixing data across embodiments helps or hinders, together with a bias-reduced real-robot evaluation protocol.

III. Methodology and Evaluation Protocol

To study how vision–language–action (VLA) models scale under heterogeneous training data, we build (i) a controlled testbed for generalist architectures and (ii) a rigorous physical evaluation protocol. In this section, we describe the model, the cross-embodiment unification used for physical evaluation, and a bias-resistant testing procedure, as illustrated in Figure 1.

A. Mixture-of-Transformers with Flow-Matching Control

Our goal is to combine high-frequency, precise control with strong semantic reasoning. Following prior work, we adopt a Mixture-of-Transformers (MoT) design. [10, 3].

Dual Experts with Layerwise Shared Attention. The model uses two transformer backbones in parallel. The **Semantic Expert** ($\mathcal{E}_{\text{sem}}$) is initialized from a pretrained VLM to preserve visual and language priors. The **Action Expert** ($\mathcal{E}_{\text{act}}$) is a randomly initialized and trained for control. For efficiency, it uses a smaller hidden dimension, but it matches the Semantic Expert in depth. We split the input sequence into four token groups: $\mathbf{X} = \{\mathbf{X}_V, \mathbf{X}_L, \mathbf{X}_S, \mathbf{X}_A\}$, corresponding to visual tokens, instruction tokens, proprioceptive state tokens, and action-latent tokens. The Semantic Expert processes the

$\{\mathbf{X}_V, \mathbf{X}_L\}$, while the Action Expert is dedicated to the kinematic subset $\{\mathbf{X}_S, \mathbf{X}_A\}$. To support strong cross-modal fusion, the two experts interact at every layer through a shared causal self-attention mechanism. Although the hidden sizes of the two streams differ, we use the same number of attention heads and the same per-head dimension in both experts. At each layer, we compute attention by concatenating the projected Queries, Keys, and Values from the two streams as follows:

$$\mathbf{T}^{(l)} = \left[\, \mathbf{T}^{(l)}_{\text{sem}} \;\; \mathbf{T}^{(l)}_{\text{act}} \,\right], \quad \mathbf{T} \in \{\mathbf{Q}, \mathbf{K}, \mathbf{V}\}, \tag{1}$$

where square brackets $[\,\cdot\,]$ denote concatenation along the sequence dimension. This lets the Action Expert attend to the full visual–semantic context directly, injecting high-frequency control information without first compressing it through a discrete text-embedding space.

Continuous Action Chunking via Flow Matching. We use flow matching [24] to model action generation as a conditional distribution over an action chunk $p(\mathbf{a}_{t:t+H}|\mathbf{o}_t)$ where H is a short prediction horizon. This formulation naturally produces smooth and potentially action trajectories. Concretely, the Action Expert learns a time-dependent vector field $v_t(\mathbf{x}, \tau)$ that transports samples from a Gaussian noise distribution $\pi_0(\mathbf{x}) = \mathcal{N}(\mathbf{0}, \mathbf{I})$ to the action-chunk data distribution $\pi_1(\mathbf{x}) \approx \mathbf{a}_{t:t+H}$, conditioned on multimodal context c (Semantic Expert features and proprioception). During training, given a data sample $\mathbf{x}_1$, a noise sample $\mathbf{x}_0 \sim \pi_0$, and a timestep $\tau \sim \mathcal{U}[0, 1]$, we form the interpolated point $\mathbf{x}_\tau = (1 - \tau)\mathbf{x}_0 + \tau\mathbf{x}_1$ and minimize the flow-matching objective:

$$\mathcal{L}_{\text{FM}} = \mathbb{E}_{\tau, \mathbf{x}_0, \mathbf{x}_1}\left[\left\| v_\theta(\mathbf{x}_\tau, \tau, \mathbf{c}) - (\mathbf{x}_1 - \mathbf{x}_0) \right\|^2\right], \tag{2}$$

where $\mathbf{c}$ denotes the multimodal conditioning context combining features from the Semantic Expert with proprioceptive inputs. At inference time, we integrate the resulting Ordinary Differential Equation (ODE) with an explicit Euler solver to generate the action sequence.

Physically Grounded Unified Action Space. To support scalable pre-training across heterogeneous robots—from single-arm grippers to bimanual dexterous hands—we define a physically grounded unified action space $\mathcal{A}_{\text{uni}} \in \mathbb{R}^D$. This space acts as a superset of all supported physical degrees of freedom (DoF), and is partitioned into semantically aligned subspaces:

$$\mathcal{A}_{\text{uni}} = \mathcal{S}_{\text{eef}} \oplus \mathcal{S}_{\text{joint}} \oplus \mathcal{S}_{\text{gripper}} \oplus \mathcal{S}_{\text{hand}} \oplus \mathcal{S}_{\text{aux}} \tag{3}$$

Here, $\mathcal{S}_{\text{eef}}$ represents bimanual end-effector poses (translation and axis–angle rotation) for Cartesian control; $\mathcal{S}_{\text{joint}}$ supports direct joint-space commands (e.g., 7-DoF arms); $\mathcal{S}_{\text{gripper}}$ encodes parallel-jaw gripper states; $\mathcal{S}_{\text{hand}}$ provides higher-dimensional slots for dexterous hands; and $\mathcal{S}_{\text{aux}}$ covers auxiliary mechanisms. To study which action parameterization scales best, we allow flexible action parameterization within $\mathcal{S}_{\text{eef}}$. Let $\mathbf{T}_\tau \in SE(3)$ denote the target end-effector pose at horizon step τ, and $\mathbf{T}_0$ be the pose at the start of the action chunk. We define a coordinate mapping Ψ with five modes:

$$\Psi(\mathbf{T}_\tau) \in \begin{cases} \mathbf{T}_\tau \ominus \mathbf{T}_0 & \text{(World-Rel)} \\ \mathbf{T}_\tau \ominus \mathbf{T}_{\tau-1} & \text{(World-Delta)} \\ \mathbf{T}_0^{-1} \circ \mathbf{T}_\tau & \text{(EEF-Rel)} \\ \mathbf{T}_{\tau-1}^{-1} \circ \mathbf{T}_\tau & \text{(EEF-Delta)} \end{cases} \tag{4}$$

where $\ominus$ computes the pose difference in the world frame (translation difference and rotational displacement), and $\circ$ denotes $SE(3)$ composition in the local (end-effector) frame. *World/EEF-Rel* express displacement relative to the chunk start, whereas *World/EEF-Delta* capture step-to-step increments. For each robot embodiment r with native action space $\mathcal{A}_r$, we define an embedding mapping $\phi_r : \mathcal{A}_r \to \mathcal{A}_{\text{uni}}$ that places robot-specific actions into the appropriate semantic slots of the unified space. Dimensions not used by embodiment r are disabled with a binary mask. This construction encourages the model to learn shared physical priors in subspaces that overlap across robots, while still allowing embodiment-specific control in non-overlapping subspaces.

Algorithm 1 Grouped Blind Ensemble Protocol

Require: Model Pool $\mathcal{M}$, Task Set $\mathcal{T}$, Group Size K, Trials per model N

Divide $\mathcal{M}$ into random non-overlapping groups $\{G_1, \ldots, G_M\}$
for each task $\tau \in \mathcal{T}$ **do**
  for each model group G_j **do**
    $\mathcal{H} \leftarrow$ Anonymize(G_j) ▷ Map models to aliases
    $\mathcal{Q} \leftarrow$ Shuffle$(\{\underbrace{h_1, \ldots, h_1}_{N}, \ldots, \underbrace{h_K, \ldots, h_K}_{N}\})$ ▷ Create randomized trial queue
    Initialize results $\mathcal{R}_j \leftarrow \emptyset$
    while $\mathcal{Q}$ is not empty **do**
      $h \leftarrow \mathcal{Q}$.pop() ▷ Get next anonymous model
      Operator executes task τ with policy π_h
      Record outcome $r \in \{0, 1\}$ to $\mathcal{R}_j$
    end while
    Break ▷ Allow operator rest between groups
  end for
end for
return Deanonymized aggregate statistics

B. Grouped Blind Ensemble Evaluation

Evaluating robotic foundation models is susceptible to experimenter bias. To enable objective comparisons across training strategies, we propose the Grouped Blind Ensemble Protocol, which implements a double-blind procedure that cleanly separates policy execution (inference) from outcome assessment. As shown in Algorithm 1, we randomly partition the model pool into manageable groups of size K (typically 4-8). For each task, models within the active group are anonymized and evaluated in a randomized order. The system dispatches the next policy to run, and the operator functions only as an executor by performing the rollout and recording

binary success/failure without access to model identities or versions. Grouping serves two purposes: it reduces the influence of human preferences by keeping the operator blind to the tested model, and it supports structured breaks between groups to mitigate fatigue, helping maintain consistent evaluation quality in large-scale studies.

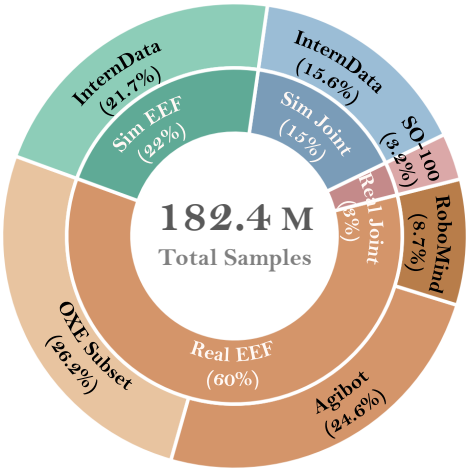

Fig. 2: Composition of the balanced pre-training data.

IV. Pre-training Data and Implementation

In this section, we describe how we construct our heterogeneous corpus for VLA pre-training and summarize the key implementation settings. We highlight two ingredients that are essential for stable learning at scale: balancing data contributions across domains with very different sizes, and applying regularization to improve robustness and generalization.

A. Large-scale Heterogeneous Robot Data

Training a generalist VLA requires data that spans diverse embodiments, environments, and control modalities. We therefore aggregate a large-scale collection of robot manipulation trajectories and organize the sources along two axes: domain (real vs. simulation) and control space (end-effector vs. joint space). This yields four quadrants, which we use to structure and report the composition of our training mixture as follows.

Real-World End-Effector Data. Real-world end-effector (EEF) trajectories form the core of our manipulation pre-training data. We include the Open X-Embodiment (OXE) dataset [30], only using its high-quality subsets such as DROID, Bridge, and Fractal. We further incorporate large-scale proprietary datasets from Agibot [5], which cover both gripper-based and dexterous-hand EEF control, and RoboMind [39, 15], spanning multiple platforms including Franka, UR5, and AgileX mobile manipulators.

Simulation and Joint-Space Data. To broaden kinematic coverage and improve fine-grained control, we incorporate simulation trajectories from InternData [6], including both conventional arm manipulation and dexterous-hand tasks. To strengthen joint-space modeling, we additionally use data from low-cost teleoperation platforms such as SO-100 [33] and real-world humanoid datasets [35].

Data Balancing Strategy. A key challenge in co-training is the severe imbalance in dataset scale: raw frame counts range from hundreds of millions to only a few hundred thousand. Since most existing VLA training treats each frame transition as an independent sample, naively mixing datasets would cause the largest sources to dominate the gradient. We mitigate this issue by applying a dynamic downsampling strategy using a dataset-specific via `frame_step_size`. Concretely, we aggressively downsample dense simulation streams and high-frequency real-world logs (e.g., step size 7 for Agibot Gripper; step sizes 4-8 for InternData Sim subsets), while keeping dense sampling for smaller but diverse datasets (e.g., step size 1 for SO-100 and most OXE subsets). After balancing, the effective corpus contains approximately 180 million frame transitions. Figure 2 illustrates the balanced distribution of the pre-training mixture.

B. Pretraining Implementation Details

For model configuration, the semantic expert $\mathcal{E}_{\text{sem}}$ is initialized from InternVL-3.5-2B [38] with hidden size 2048, and the action expert $\mathcal{E}_{\text{act}}$ is a Transformer decoder trained from scratch, comprising 0.7B parameters with hidden size 1024. To enable the synchronized attention mechanism, we use a shared per-head dimension in both experts, ensuring that their Query/Key/Value projections can be concatenated despite the mismatch in hidden sizes. The model operates in the unified action space using relative end-effector (EEF) commands for Cartesian control and absolute values for joint space control. To assess the effect of regularization in large-scale VLA pre-training, we integrate two widely used stochastic techniques into our training pipeline.

During pretraining, we first mask proprioception by zeroing the input proprioceptive vector with probability $p = 0.2$. Then, we also apply independent dropping of each camera view with $p = 0.2$. Note that at least one view is retained if all are stochastically dropped to ensure valid visual inputs. We use a two-stage curriculum to investigate whether gradual adaptation outperforms direct joint optimization. In Stage 1, we freeze the VLM backbone and optimize only the action expert for 40k steps. In Stage 2, we unfreeze the full model and train all parameters for 200k steps. All experiments use a global batch size of 256 on 8 NVIDIA A800 GPUs.

V. Experiments

In this section, we present a systematic study that disentangles the scaling recipe for VLAs trained on heterogeneous robot data. Our analysis centers on three key design dimensions: **(1) Physical Alignment** which investigates how coordinate frames and action representations affect robot control precision; **(2) Embodiment Mixture** which assess whether aggregating heterogeneous data sources yields positive transfer or introduces interference instead; and **(3) Training Regularization** which evaluates the effectiveness of sensory dropout strategies at scale.

TABLE I: Pre-training transfer ablation on Libero 5-shot benchmark under different action space configuration.

Action Space	Init.	Unfrozen VLM					Frozen VLM				
		Spatial	Object	Goal	Long	**Avg**	Spatial	Object	Goal	Long	**Avg**
World-Frame Coordinates											
World-Relative	Scratch	90.8	93.6	78.2	74.8	84.4	72.2	85.0	75.4	43.6	69.1
	Pretrain	89.8 (-1.0)	86.8 (-6.8)	88.4 (+10.2)	68.8 (-6.0)	83.5 (-0.9)	81.2 (+9.0)	89.8 (+4.8)	75.0 (-0.4)	52.6 (+9.0)	74.7 (+5.6)
World-Delta	Scratch	90.4	93.0	82.8	73.6	85.0	74.2	75.0	75.0	42.6	66.7
	Pretrain	86.2 (-4.2)	92.0 (-1.0)	88.8 (+6.0)	71.0 (-2.6)	84.5 (-0.5)	82.6 (+8.4)	89.6 (+14.6)	73.8 (-1.2)	53.2 (+10.6)	74.8 (+8.1)
End-Effector Coordinates											
EEF-Relative	Scratch	87.6	93.0	76.4	70.6	81.9	73.0	84.2	70.6	39.6	66.9
	Pretrain	86.0 (-1.6)	90.0 (-3.0)	88.6 (+12.2)	73.4 (+2.8)	84.5 (+2.6)	79.8 (+6.8)	90.6 (+6.4)	76.6 (+6.0)	53.2 (+13.6)	75.1 (+8.2)
EEF-Delta	Scratch	87.6	93.0	72.0	67.0	79.9	73.2	79.8	70.4	41.0	66.1
	Pretrain	85.6 (-2.0)	89.8 (-3.2)	87.4 (+15.4)	66.2 (-0.8)	82.3 (+2.4)	77.8 (+4.6)	88.2 (+8.4)	73.6 (+3.2)	48.4 (+7.4)	72.0 (+5.9)

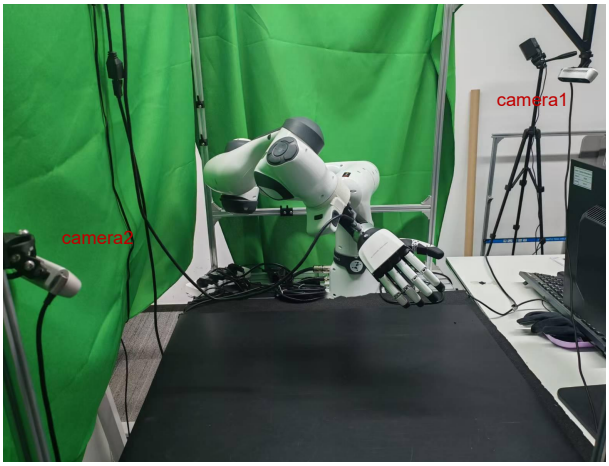

Fig. 3: Real-world experimental setup.

Action Space	Init.	Pick/ Place	Doors/ Drawers	Other	**Avg**
World-Frame Coordinates					
World-Relative	Scratch	20.0	60.0	40.0	38.3
	Pretrain	22.5 (+2.5)	58.3 (-1.7)	45.0 (+5.0)	40.8 (+2.5)
World-Delta	Scratch	21.0	62.3	46.0	41.8
	Pretrain	23.8 (+2.8)	60.0 (-2.3)	45.0 (-1.0)	41.7 (-0.1)
End-Effector Coordinates					
EEF-Relative	Scratch	33.0	57.3	47.4	45.1
	Pretrain	35.3 (+2.3)	62.3 (+5.0)	54.4 (+7.0)	50.0 (+4.9)
EEF-Delta	Scratch	26.0	63.7	44.8	43.3
	Pretrain	36.3 (+10.3)	56.7 (-7.0)	49.0 (+4.2)	46.7 (+3.4)

TABLE II: Pre-training transfer ablation on RoboCasa benchmark under different action space configuration.

Our evaluation is designed to rigorously stress-test the quality of the pre-trained representations in both standardized simulation benchmarks and real-world deployments.

Simulation Benchmarks. We first evaluate downstream transfer on LIBERO [25] and RoboCasa [28]. In both benchmarks, we follow a multi-task fine-tuning protocol in which a single unified policy is trained jointly across all tasks. For LIBERO, we fine-tune on the union of all four task suites (Spatial, Object, Goal, Long) under a low-data regime of 5 demonstrations per task (5-shot). For RoboCasa, we fine-tune on all 24 kitchen tasks using the standard few-shot setup of 50 human demonstrations per task.

Real-Robot Protocol. As shown in Figure 3, our experimental setup features a Franka Panda arm equipped with two RGB cameras positioned on the left and right sides. We evaluate four tasks that probe complementary capabilities. *Stack Bowls* focuses on precision with a maximum score of 3; the task requires placing three bowls into a plate, and we award 1 point for each successful placement. *Pick-to-Drawer* examines long-horizon planning with a maximum score of 5; the score comprises 1 point for opening the drawer, 1 point for each of the three objects placed inside, and 1 point for closing the drawer. *Wipe Board* involves dynamic motion with a maximum score of 4; we assign 1 point for each specific step: picking up the cloth, initiating the wiping motion, wiping the surface clean, and placing the cloth back. *Water Plant* targets non-rigid object interaction with a maximum score of 3; the sequence includes picking up the spray can, aiming at the plant, and pressing the trigger, where each action contributes 1 point. For each task, we conduct 10 trials, calculate the total score, and multiply it by a fixed coefficient to obtain a percentage. During inference, we adopt a grouped blind ensemble protocol: for each experimental setting, the operator remains unaware of the model variant to minimize subjective bias.

A. Exploration of Physical Alignment

A key challenge in leveraging heterogeneous robot data is to standardize the configuration of action representation for robots with different kinematics and base placements. To study this, we evaluate four coordinate parameterizations: **World-Relative**, **World-Delta**, **EEF-Relative**, and **EEF-Delta**, and compare policies trained from scratch with those fine-tuned from our heterogeneous pre-training corpus.

Simulation Results. Table I shows that the coordinate frame strongly influences the performance of pre-training transfer. On LIBERO, scratch-trained policies paradoxically perform

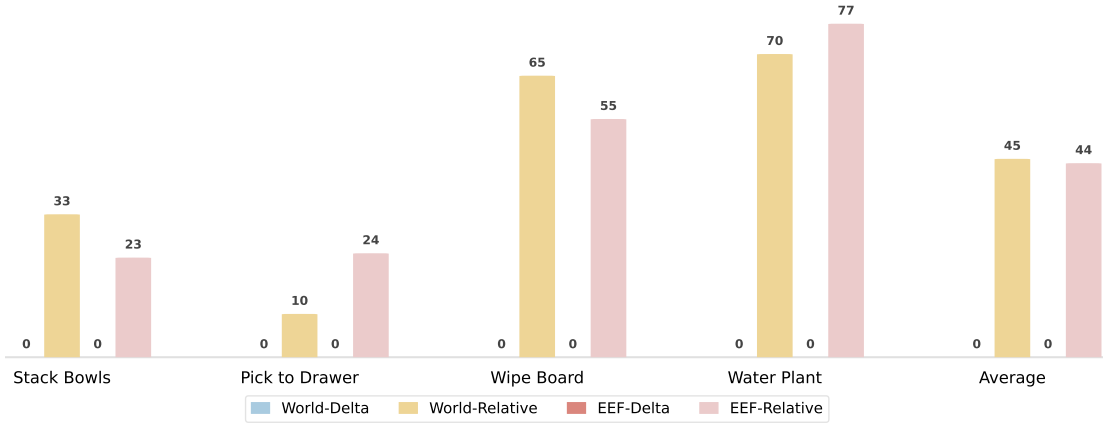

Fig. 4: Real-world blind evaluation of different action spaces.

TABLE III: LIBERO 5-shot transfer performance across incremental pre-training mixtures.

Pre-train Mixture	**Frozen VLM**					**Unfrozen VLM**				
	Spat	Obj	Goal	Long	**Avg**	Spat	Obj	Goal	Long	**Avg**
Scratch	73.0	84.2	70.6	39.6	66.9	87.6	93.0	76.4	70.6	81.9
$\mathcal{D}_1$: OXE Only	**87.6** (+14.6)	88.2 (+4.0)	78.8 (+8.2)	**54.6** (+15.0)	**77.3** (+10.4)	**89.2** (+1.6)	89.2 (-3.8)	**92.8** (+16.4)	**74.4** (+3.8)	**86.4** (+4.5)
⌞ $\mathcal{D}_2$: $\mathcal{D}_1$ + Real EEF	84.4 (-3.2)	86.0 (-2.2)	77.4 (-1.4)	47.4 (-7.2)	73.8 (-3.5)	86.2 (-3.0)	**94.6** (+5.4)	88.8 (-4.0)	66.4 (-8.0)	84.0 (-2.4)
⌞ $\mathcal{D}_3$: $\mathcal{D}_2$ + Sim EEF	76.0 (-8.4)	82.2 (-3.8)	**79.0** (+1.6)	51.2 (+3.8)	72.1 (-1.7)	87.4 (+1.2)	91.4 (-3.2)	86.2 (-2.6)	69.6 (+3.2)	83.7 (-0.3)
⌞ $\mathcal{D}_4$: $\mathcal{D}_3$ + Joint	79.8 (+3.8)	**90.6** (+8.4)	76.6 (-2.4)	53.2 (+2.0)	75.1 (+3.0)	86.0 (-1.4)	90.0 (-1.4)	88.6 (+2.4)	73.4 (+3.8)	84.5 (+0.8)

Pre-train Mixture	Pick/ Place	Doors/ Drawers	Other	**Avg**
Scratch	33.0	57.3	47.4	45.1
$\mathcal{D}_1$: OXE Only	**42.0** (+9.0)	**72.3** (+15.0)	54.2 (+6.8)	**54.7** (+9.6)
⌞ $\mathcal{D}_2$: $\mathcal{D}_1$ + Real EEF	38.5 (-3.5)	60.5 (-11.8)	47.4 (-6.8)	48.8 (-5.9)
⌞ $\mathcal{D}_3$: $\mathcal{D}_2$ + Sim EEF	36.3 (-2.2)	56.7 (-3.8)	**56.0** (+8.6)	49.6 (+0.8)
⌞ $\mathcal{D}_4$: $\mathcal{D}_3$ + Joint	35.3 (-1.0)	62.3 (+5.6)	54.4 (-1.6)	50.0 (+0.4)

TABLE IV: RoboCasa 50-shot transfer performance across incremental pre-training mixtures.

better with world-frame actions compared to using EEF coordinates, likely because fixed camera extrinsics and bounded workspace make absolute positions easy to exploit. In contrast, pre-training yields negative transfer for world-frame variants (-0.9%, -0.5%) but consistent gains for EEF-frame variants (+2.6%, +2.4%). This inversion indicates that world coordinates can overfit single-environment regularities, whereas EEF coordinates generalize better across varying cameras and robot bases in large-scale data.

The benefits of pre-training become even more pronounced when the VLM backbone is frozen. In this regime, the VLM backbone remains fixed during fine-tuning, the performance relies heavily on the high-quality representations initialized from pre-training, including both the aligned VLM and the Action Expert. We observe substantial gains over the scratch baseline across all action spaces. Most notably, EEF-Relative achieves the highest average success rate of 75.1% with the largest improvement of +8.2%. This confirms that the end-effector space is the most effective choice for preserving and transferring physical priors when the visual backbone is static.

These trends are even more pronounced on the geometrically diverse RoboCasa benchmark (Table II). The EEF-Relative representation exhibits the most reliable scaling behavior, increasing average success from 45.1% to 50.0%. Gains are consistent across task categories, including a +5.0% improvement on articulated-object tasks (Doors/Drawers) and a +7.0% increase on the remaining tasks. In contrast, alternative action representations are unstable or ineffective. For instance, World-Delta shows essentially no benefit and even slight negative transfer (-0.1%), while degrades substantially on Doors/Drawers (-7.0%) despite pretraining.

Real-World Results. Beyond simulation, we further corroborate these findings with blinded real-world evaluations (Figure 4). We note that all EEF-Delta actions exhibit jittering in place on the real robot, rendering them unable to execute tasks and resulting in a 0% success rate. In contrast, Relative actions execute effectively. We observe no significant performance difference between World-Relative and EEF-Relative in real-world settings.

B. Exploration of Embodiment Mixture

With the action space fixed to EEF-Relative, we study how performance changes as we scale heterogeneous pre-training data using a cumulative inclusion protocol. Specifically, we form four progressively richer data mixtures to isolate the

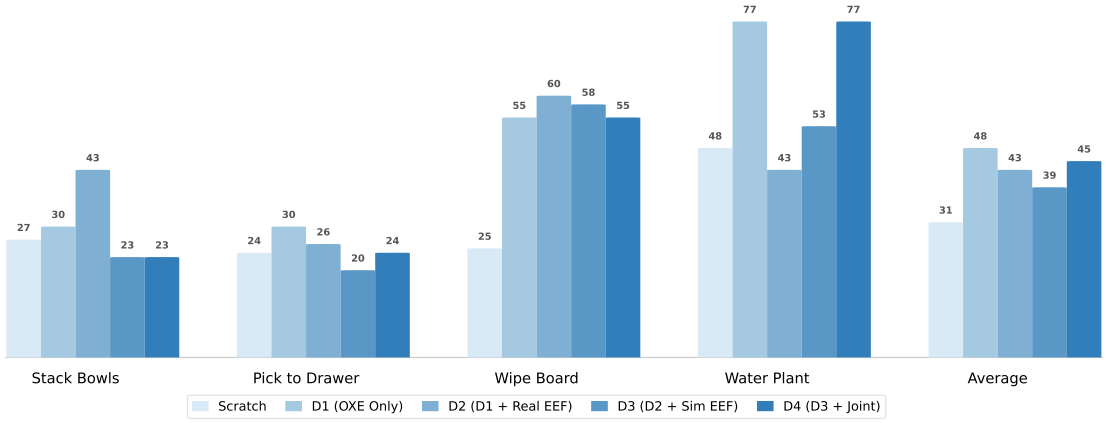

Fig. 5: Real-world blind evaluation of different pre-training data mixtures.

marginal contribution of each data source: **D1 (OXE Only)** utilizes standard public datasets. **D2 (+Real EEF)** adds real-world end-effector trajectories from additional robot embodiments. **D3 (+Sim EEF)** further includes simulation end-effector data. Finally, **D4 (+Joint)** additionally incorporates joint-space demonstrations, projected into the unified model.

In contrast to language-model scaling behavior, enlarging the robotic pre-training corpus does **NOT** yield monotonic gains. As shown in Table III, $\mathcal{D}_1$ provides the strongest pre-training transfer on LIBERO, reaching 77.3% average success in the Frozen VLM setting (+10.4% over training from scratch). However, simply increasing data diversity degrades performance. The addition of heterogeneous real-robot EEF data ($\mathcal{D}_2$) drops success to 73.8%, and including simulated EEF trajectories ($\mathcal{D}_3$) further reduces it to 72.1%. Projecting joint-space data in $\mathcal{D}_4$ recovers part of the loss (+3.0%), but still does not surpass the OXE-only baseline. This negative trend is even more pronounced on RoboCasa (Table IV). $\mathcal{D}_1$ sets a high-water mark of 54.7%, while adding diverse real-world EEF data in $\mathcal{D}_2$ immediately decreases performance to 48.8%. Later mixtures ($\mathcal{D}_3, \mathcal{D}_4$) offer only marginal recovery, leaving the final model nearly **5%** below $\mathcal{D}_1$ overall and **10.0%** worse on articulated-object tasks (Doors/Drawers). Collectively, these results suggests thatnaively pooling structurally disparate robot datasets induces destructive interference rather than improved transfer.

Real-World Results. As shown in Figure 5. Real-world experimental results demonstrate that pre-training models offer a significant advantage, as all pre-training variants (D_1 through D_4) substantially outperform the training-from-scratch (Scratch) baseline. Specifically, while the D_1 scheme using only the OXE dataset maintains robust performance across multiple tasks, the introduction of simulated end-effector (Sim EEF) trajectories in the D_3 stage leads to varying degrees of performance degradation in tasks such as "Stack Bowls," "Pick to Drawer," and "Wipe Board," further validating the potential destructive interference caused by heterogeneous data.

TABLE V: Impact of training regularization and scheduling on LIBERO 5-shot performance.

Mask Ratio		**Schedule**	**Success Rate (%)**				
State	View		Spat	Obj	Goal	Long	**Avg**
0.2	0.2	2-Stage	86.0	90.0	88.6	73.4	84.5
0	0.2	2-Stage	88.2	**96.0**	87.6	69.0	85.2
0.5	0.2	2-Stage	**88.8**	90.4	86.8	70.0	84.0
0.2	0	2-Stage	87.0	92.2	**89.2**	**74.0**	85.6
0.2	0.5	2-Stage	85.8	93.2	86.6	72.2	84.5
0.2	0.2	Stage 2 Only	88.2	93.8	87.0	**74.0**	**85.8**

C. Exploration of Training Regularization

We evaluate how sensory dropout ($p_{\text{state}}, p_{\text{view}}$) and staged training curricula affect model generalization. Results in Table V call into question the need for these widely used regularization practices at scale.

Limited Benefit from Explicit Regularization. Contrary to common assumptions, stochastic modality dropout does not provide a consistent improvement. Disabling visual dropout ($p_{\text{view}} = 0$) increases success to 85.6% versus the balanced baseline (84.5%), whereas heavy proprioceptive masking reduces performance. This pattern may suggest that the natural diversity of the pre-training corpus already serves as an effective regularizer, making additional noise injection unnecessary, or even harmful.

Direct Optimization is Sufficient The two-stage alignment curriculum is similarly non-essential. Directly fine-tuning the full model end-to-end ("Stage 2 Only") attains the best average success 85.8%, outperforming the multi-stage schedule. This indicates that the initialized action expert can quickly co-adapt with the VLM backbone under joint training, supporting a simpler and more efficient optimization pipeline.

D. Comparison with Representative Generalist Policies.

To validate our testbed, we benchmark our standard implementation against against representative generalist VLA policies (π_0 [3], $\pi_{0.5}$ [16] and GR00T-N1 [29]) under a

TABLE VI: Benchmarking against representative generalist policies on LIBERO and RoboCasa.

Model	LIBERO					RoboCasa			
	Spatial	Object	Goal	Long	**Avg**	Pick/Place	Doors/Drawers	Other	**Avg**
GR00T-N1	94.4	97.6	93.0	90.6	93.9	18.6	50.2	39.1	36.0
π_0	98.0	96.8	94.4	88.4	94.4	14.0	53.1	**58.5**	42.4
$\pi_{0.5}$	**98.8**	**98.2**	98.0	92.4	96.9	21.5	57.8	44.9	41.4
Ours (Base)	98.4	96.2	**98.4**	**98.2**	**97.9**	**35.3**	**62.3**	54.4	**50.0**

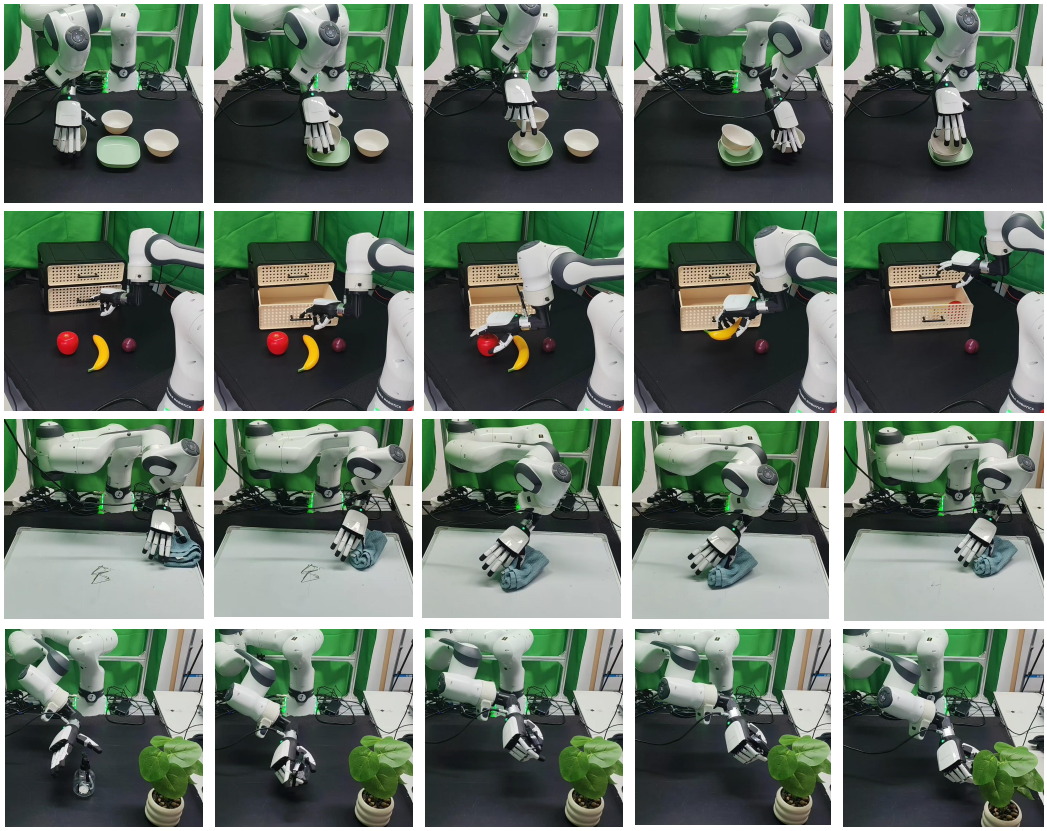

Fig. 6: Qualitative rollout sequences across diverse real-world tasks.

matched 50-shot fine-tuning protocol. As reported in Table VI, our base model is competitive without task-specific tuning or specialized optimizations, reaching 97.9% average success on LIBERO and 50.0% on RoboCasa benchmarks. These results indicate that our platform provides a strong and representative foundation, so our experimental conclusions reflect behavior in a high-performance regime aligned with current advances in robot learning.

E. Qualitative Analysis

Figure 6 presents rollout sequences for four real-world tasks in our blind evaluation protocol. The first row shows the "Stack Bowls" task which focuses on precision, and the challenge lies in the precise consecutive grasping and stacking of three bowls. The second row is the "Pick-to-Drawer" task which requires long-horizon planning to complete multiple subtasks (opening, placing, closing), and it is easy to fail in intermediate stages. The third row is the "Wipe Board" task involving dynamic motion and surface contact, which is prone to incomplete wiping. The fourth row is the "Water Plant" task targeting non-rigid object interaction, where the main difficulties are knocking over the bottle and failing to press the trigger accurately.

VI. Conclusion

In this work, we present a systematic study on scaling Vision-Language-Action (VLA) models with heterogeneous robot data. To ensure reliable evaluation, we introduce a Grouped Blind Ensemble protocol that minimizes human bias in real-world experiments. Our results show that simply increasing data scale does not automatically lead to better performance. First, the EEF-Relative action space proves to be the most effective choice for handling diverse robot kinematics. Second, scaling across different robots is difficult; simply

mixing diverse data often reduces performance, indicating that careful data alignment is essential. Finally, complex training techniques like sensory dropout do not consistently bring improvements, suggesting that simple training recipes are often sufficient. Overall, this study offers practical guidance for effectively training general-purpose VLA models.

REFERENCES

[1] Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, Mei Li, Kaixin Li, Zicheng Lin, Junyang Lin, Xuejing Liu, Jiawei Liu, Chenglong Liu, Yang Liu, Dayiheng Liu, Shixuan Liu, Dunjie Lu, Ruilin Luo, Chenxu Lv, Rui Men, Lingchen Meng, Xuancheng Ren, Xingzhang Ren, Sibo Song, Yuchong Sun, Jun Tang, Jianhong Tu, Jianqiang Wan, Peng Wang, Pengfei Wang, Qiuyue Wang, Yuxuan Wang, Tianbao Xie, Yiheng Xu, Haiyang Xu, Jin Xu, Zhibo Yang, Mingkun Yang, Jianxin Yang, An Yang, Bowen Yu, Fei Zhang, Hang Zhang, Xi Zhang, Bo Zheng, Humen Zhong, Jingren Zhou, Fan Zhou, Jing Zhou, Yuanzhi Zhu, and Ke Zhu. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025.

[2] Lars Berscheid, Pascal Meißner, and Torsten Kröger. Robot learning of shifting objects for grasping in cluttered environments. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 612–618. IEEE, 2019.

[3] Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, et al. π_0: A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 2024.

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

[5] Qingwen Bu, Jisong Cai, Li Chen, Xiuqi Cui, Yan Ding, Siyuan Feng, Shenyuan Gao, Xindong He, Xuan Hu, Xu Huang, et al. Agibot world colosseo: A large-scale manipulation platform for scalable and intelligent embodied systems. *arXiv preprint arXiv:2503.06669*, 2025.

[6] Xinyi Chen, Yilun Chen, Yanwei Fu, Ning Gao, Jiaya Jia, Weiyang Jin, Hao Li, Yao Mu, Jiangmiao Pang, Yu Qiao, et al. Internvla-m1: A spatially guided vision-language-action framework for generalist robot policy. *arXiv preprint arXiv:2510.13778*, 2025.

[7] Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, 44(10-11):1684–1704, 2025.

[8] Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

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

[10] Chaorui Deng, Deyao Zhu, Kunchang Li, Chenhui Gou, Feng Li, Zeyu Wang, Shu Zhong, Weihao Yu, Xiaonan Nie, Ziang Song, et al. Emerging properties in unified multimodal pretraining. *arXiv preprint arXiv:2505.14683*, 2025.

[11] Hao-Shu Fang, Hongjie Fang, Zhenyu Tang, Jirong Liu, Chenxi Wang, Junbo Wang, Haoyi Zhu, and Cewu Lu. Rh20t: A comprehensive robotic dataset for learning diverse skills in one-shot. *arXiv preprint arXiv:2307.00595*, 2023.

[12] Yicheng Feng, Yijiang Li, Wanpeng Zhang, Sipeng Zheng, Hao Luo, Zihao Yue, and Zongqing Lu. Videoorion: Tokenizing object dynamics in videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 20401–20412, 2025.

[13] Luo Hao, Yue Zihao, Zhang Wanpeng, Feng Yicheng, Zheng Sipeng, Ye Deheng, and Lu Zongqing. OpenMMEgo: Enhancing egocentric understanding for LMMs with open weights and data. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

[14] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[15] Chengkai Hou, Kun Wu, Jiaming Liu, Zhengping Che, Di Wu, Fei Liao, Guangrun Li, Jingyang He, Qiuxuan Feng, Zhao Jin, et al. Robomind 2.0: A multimodal, bimanual mobile manipulation dataset for generalizable embodied intelligence. *arXiv preprint arXiv:2512.24653*, 2025.

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

[17] Michael Janner, Yilun Du, Joshua B Tenenbaum, and Sergey Levine. Planning with diffusion for flexible behavior synthesis. *arXiv preprint arXiv:2205.09991*, 2022.

[18] Tao Jiang, Tianyuan Yuan, Yicheng Liu, Chenhao Lu, Jianning Cui, Xiao Liu, Shuiqi Cheng, Jiyang Gao, Huazhe Xu, and Hang Zhao. Galaxea open-world dataset and g0 dual-system vla model. *arXiv preprint arXiv:2509.00576*, 2025.

[19] Alexander Khazatsky, Karl Pertsch, Suraj Nair, Ash-

win Balakrishna, Sudeep Dasari, Siddharth Karamcheti, Soroush Nasiriany, Mohan Kumar Srirama, Lawrence Yunliang Chen, Kirsty Ellis, et al. Droid: A large-scale in-the-wild robot manipulation dataset. *arXiv preprint arXiv:2403.12945*, 2024.

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

[21] Qixiu Li, Yaobo Liang, Zeyu Wang, Lin Luo, Xi Chen, Mozheng Liao, Fangyun Wei, Yu Deng, Sicheng Xu, Yizhong Zhang, et al. Cogact: A foundational vision-language-action model for synergizing cognition and action in robotic manipulation. *arXiv preprint arXiv:2411.19650*, 2024.

[22] Zhiqi Li, Guo Chen, Shilong Liu, Shihao Wang, Vibashan VS, Yishen Ji, Shiyi Lan, Hao Zhang, Yilin Zhao, Subhashree Radhakrishnan, et al. Eagle 2: Building post-training data strategies from scratch for frontier vision-language models. *arXiv preprint arXiv:2501.14818*, 2025.

[23] Zhixuan Liang, Yizhuo Li, Tianshuo Yang, Chengyue Wu, Sitong Mao, Tian Nian, Liuao Pei, Shunbo Zhou, Xiaokang Yang, Jiangmiao Pang, et al. Discrete diffusion vla: Bringing discrete diffusion to action decoding in vision-language-action policies. *arXiv preprint arXiv:2508.20072*, 2025.

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

[25] Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. Libero: Benchmarking knowledge transfer for lifelong robot learning. *Advances in Neural Information Processing Systems*, 36:44776–44791, 2023.

[26] Songming Liu, Lingxuan Wu, Bangguo Li, Hengkai Tan, Huayu Chen, Zhengyi Wang, Ke Xu, Hang Su, and Jun Zhu. Rdt-1b: a diffusion foundation model for bimanual manipulation. *arXiv preprint arXiv:2410.07864*, 2024.

[27] Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003*, 2022.

[28] Soroush Nasiriany, Abhiram Maddukuri, Lance Zhang, Adeet Parikh, Aaron Lo, Abhishek Joshi, Ajay Mandlekar, and Yuke Zhu. Robocasa: Large-scale simulation of everyday tasks for generalist robots. *arXiv preprint arXiv:2406.02523*, 2024.

[29] J Bjorck Nvidia, Fernando Castaneda, 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. *arXiv preprint arXiv:2503.14734*, 2025.

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

[31] Delin Qu, Haoming Song, Qizhi Chen, Yuanqi Yao, Xinyi Ye, Yan Ding, Zhigang Wang, JiaYuan Gu, Bin Zhao, Dong Wang, et al. Spatialvla: Exploring spatial representations for visual-language-action model. *arXiv preprint arXiv:2501.15830*, 2025.

[32] Nur Muhammad Mahi Shafiullah, Anant Rai, Haritheja Etukuru, Yiqian Liu, Ishan Misra, Soumith Chintala, and Lerrel Pinto. On bringing robots home. *arXiv preprint arXiv:2311.16098*, 2023.

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

[34] Andreas Steiner, André Susano Pinto, Michael Tschannen, Daniel Keysers, Xiao Wang, Yonatan Bitton, Alexey Gritsenko, Matthias Minderer, Anthony Sherbondy, Shangbang Long, et al. Paligemma 2: A family of versatile vlms for transfer. *arXiv preprint arXiv:2412.03555*, 2024.

[35] Yang Tian, Yuyin Yang, Yiman Xie, Zetao Cai, Xu Shi, Ning Gao, Hangxu Liu, Xuekun Jiang, Zherui Qiu, Feng Yuan, et al. Interndata-a1: Pioneering high-fidelity synthetic data for pre-training generalist policy. *arXiv preprint arXiv:2511.16651*, 2025.

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

[37] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024.

[38] Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, et al. Internvl3. 5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*, 2025.

[39] Kun Wu, Chengkai Hou, Jiaming Liu, Zhengping Che, Xiaozhu Ju, Zhuqin Yang, Meng Li, Yinuo Zhao, Zhiyuan Xu, Guang Yang, et al. Robomind: Benchmark on multi-embodiment intelligence normative data for robot manipulation. *arXiv preprint arXiv:2412.13877*, 2024.

[40] Shihan Wu, Xuecheng Liu, Shaoxuan Xie, Pengwei Wang, Xinghang Li, Bowen Yang, Zhe Li, Kai Zhu, Hongyu Wu, Yiheng Liu, et al. Robocoin: An open-

sourced bimanual robotic data collection for integrated manipulation. *arXiv preprint arXiv:2511.17441*, 2025.

[41] Jianglong Ye, Keyi Wang, Chengjing Yuan, Ruihan Yang, Yiquan Li, Jiyue Zhu, Yuzhe Qin, Xueyan Zou, and Xiaolong Wang. Dex1b: Learning with 1b demonstrations for dexterous manipulation. *arXiv preprint arXiv:2506.17198*, 2025.

[42] Wanpeng Zhang, Yicheng Feng, Hao Luo, Yijiang Li, Zihao Yue, Sipeng Zheng, and Zongqing Lu. Unified multimodal understanding via byte-pair visual encoding. *arXiv preprint arXiv:2506.23639*, 2025.

[43] Wanpeng Zhang, Zilong Xie, Yicheng Feng, Yijiang Li, Xingrun Xing, Sipeng Zheng, and Zongqing Lu. From pixels to tokens: Byte-pair encoding on quantized visual modalities. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=3TnLGGHhNx.

[44] Brianna Zitkovich, Tianhe Yu, Sichun Xu, Peng Xu, Ted Xiao, Fei Xia, Jialin Wu, Paul Wohlhart, Stefan Welker, Ayzaan Wahid, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. In *Conference on Robot Learning*, pages 2165–2183. PMLR, 2023.

APPENDIX

A. Data Mixture Statistics

Table VII details the composition of the balanced pre-training corpus. We calculate the effective frame count N_{eff} using the raw frame count N_{raw} and the sampling step size S:

$$N_{\text{eff}} = \lfloor N_{\text{raw}}/S \rfloor \tag{5}$$

The final balanced dataset contains approximately 182.4 million frames. We assign larger step sizes ($S \in \{3, 4, 7, 8\}$) to the massive Agibot and InternData datasets. This keeps them balanced with the Open X-Embodiment (OXE) subsets.

TABLE VII: Detailed statistics of the data mixture.

Category	Dataset Source	S	N_{raw}	N_{eff}
Real EEF	OXE - DROID	1	27.04M	27.04M
	OXE - Bridge	1	1.89M	1.89M
	OXE - BC-Z	1	5.47M	5.47M
	OXE - Language Table	1	7.05M	7.05M
	OXE - Fractal	1	3.79M	3.79M
	OXE - Kuka	1	2.46M	2.46M
	Subtotal (OXE)		*47.70M*	**47.70M**
	Agibot Gripper	7	249.04M	35.58M
	Agibot Dexterous	1	9.29M	9.29M
	RoboMind AgileX	1	5.10M	5.10M
	RoboMind UR	1	1.34M	1.34M
	RoboMind Franka	1	3.34M	3.34M
	RoboMind Tienkung (Gello)	1	2.99M	2.99M
	RoboMind Tienkung (Xsens)	1	3.18M	3.18M
	Subtotal		*321.98M*	**108.52M**
Sim EEF	InternData-M1 (Franka)	4	93.39M	23.35M
	InternData-A1 (Franka)	3	48.82M	16.27M
	Subtotal		*142.21M*	**39.62M**
Sim Joint	InternData-A1 (Lift2)	8	92.29M	11.54M
	InternData-A1 (Split-Aloha)	8	90.54M	11.31M
	InternData-A1 (Genie)	1	5.56M	5.56M
	Subtotal		*188.39M*	**28.41M**
Real Joint	SO-100	1	5.02M	5.02M
	InternData-A1 (Real Genie)	1	0.92M	0.92M
	Subtotal		*5.94M*	**5.94M**
Total			*658.52M*	**182.49M**

B. Training Hyperparameters

We conduct pre-training in two stages. In the first stage, we freeze the VLM backbone and train the action expert for 40,000 steps with a learning rate of 1×10^{-4}. In the second stage, we train the full model for 200,000 steps with a learning rate of 2×10^{-5}. Throughout pre-training, we use a global batch size of 256. We use the AdamW optimizer with a cosine learning rate scheduler, a warmup ratio of 0.05, and a weight decay of 1×10^{-5}.

For downstream fine-tuning, we update the full model with a global batch size of 128 and a learning rate of 1×10^{-4}. We train for 30,000 steps on the LIBERO 5-shot benchmark. For the full LIBERO benchmark and RoboCasa, we train for 60,000 steps.

C. Real-World Task Details

We evaluate the policies on a Franka Panda robot with two RGB cameras. Each task consists of 10 evaluation trials with randomized object initialization. We record success using the following multi-stage criteria.

- **Stack Bowls (Max Score: 3)**
  Three bowls are initialized at random positions on the tabletop. The robot must identify, grasp, and stack them into a target plate sequentially.
  - **Point 1:** Successfully grasp the first bowl and place it into the plate.
  - **Point 2:** Grasp the second bowl and stack it inside the first one.
  - **Point 3:** Grasp the third bowl and stack it inside the second one.

  Difficulty: The main difficulty is to precisely grasp and stack three bowls in a row.
- **Pick-to-Drawer (Max Score: 5)**
  This is a long-horizon task involving articulated object manipulation. The scene contains a closed drawer and three distinct objects.
  - **Point 1:** Grasp the handle and fully open the drawer.
  - **Point 2:** Pick up Object A and place it inside the drawer.
  - **Point 3:** Pick up Object B and place it inside the drawer.
  - **Point 4:** Pick up Object C and place it inside the drawer.
  - **Point 5:** Push the drawer back to a fully closed state.

  Difficulty: Since this task involves many steps, failures often occur during the intermediate stages.
- **Wipe Board (Max Score: 4)**
  This task evaluates dynamic contact-rich manipulation. A sponge is placed on the table, and the goal is to wipe a whiteboard surface.
  - **Point 1:** Successfully grasp the sponge from the table.
  - **Point 2:** Establish contact between the sponge and the board.
  - **Point 3:** Execute a continuous wiping motion.
  - **Point 4:** Return the sponge to a designated area.

  Difficulty: A common failure is that the robot does not wipe the surface completely clean.
- **Water Plant (Max Score: 3)**
  The robot interacts with a spray bottle to water a plant. This requires grasping an object with a specific functional orientation.
  - **Point 1:** Grasp the spray bottle by the handle.
  - **Point 2:** Reorient the bottle so the nozzle aims at the plant.
  - **Point 3:** Press the trigger mechanism.

  Difficulty: Common failures include knocking over the bottle or failing to press the trigger accurately.