SIMVLA 2.0: INSTRUCTION-CONDITIONED CONTROL WITHOUT LANGUAGE PRETRAINING

Anonymous authors
Paper under double-blind review

ABSTRACT

Vision-Language-Action policies typically retain pretrained language components, yet on finite, recurring task inventories the role of that retention is rarely tested directly. Action training itself can erode the competence a pretrained backbone contributes, and what remains often fails to transfer under meaning-preserving paraphrases. We therefore build the opposite end of the design space: instructions remain, but no pretrained language-side component is loaded. SIMVLA 2.0 loads no language model, text encoder, or token embedding: a fixed, parameter-free mapping indexes randomly initialized embeddings learned solely from action supervision, and a joint action Transformer conditions a continuous action head on the embeddings and the pretrained vision encoder's features. Across seven benchmark evaluations spanning six manipulation domains, it remains within the range of recent language-pretrained systems—LIBERO 96.9%, LIBERO-PRO 96.5% under semantic instruction perturbations, CALVIN ABC→D a 4.24 mean chain length, WidowX 91.7%, Google Robot 82.0%, and RoboTwin 2.0 Easy/Hard 94.0%/87.2%. On the 2025 BEHAVIOR Challenge Public Validation leaderboard it places second among Standard-track entries. Ablations separate the sources of performance: instruction conditioning is essential, pretrained token embeddings provide no consistent advantage over action-trained codes, and strong visual pretraining remains important. Code is available at https://anonymous.4open.science/r/SimVLA2-4E2F.

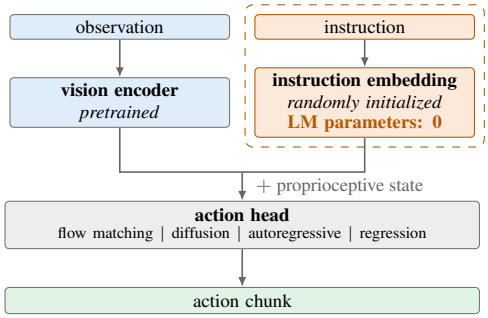
1 INTRODUCTION

Vision-Language-Action (VLA) research increasingly couples pretrained multimodal backbones with autoregressive, diffusion, or flow-matching action generators (Kim et al., 2024; Black et al., 2024; Physical Intelligence et al., 2025). This recipe is effective, but it leaves a basic attribution question unresolved. When a benchmark repeatedly draws instructions from a finite task inventory, does control require pretrained linguistic knowledge, or can the instruction act as a stable task code whose meaning is acquired from demonstrations? We approach this empirically rather than analytically: we construct a policy that is conditioned on instructions but uses no pretrained language components, and measure how far it gets.

Why the necessity question is open. The usual defense of the language-model component is linguistic generalization. A pretrained backbone should let a policy follow "pick up the banana", "grasp the banana", and "get the yellow fruit" interchangeably. Recent diagnostics arrive from two directions. First, the inherited competence does not survive action training: Zhang et al. (2026a) report that standard end-to-end VLA fine-tuning removes almost all of the backbone's multimodal understanding, and that vision–language co-training reduces but does not eliminate this "embodiment tax". Second, whatever language ability remains does not produce paraphrase generalization. Under meaning-preserving rewordings of LIBERO instructions, seven VLA configurations spanning 0.6B–7.5B and four architecture families lose 22.8–51.9 points of success, and 80–96% of those failures consist of wrong-task selection rather than imprecise execution (Kim et al., 2026). The degradation is driven mainly by substituting object words, which indicates surface-form matching rather than semantic grounding, and quadrupling task-level data diversity leaves it essentially unchanged. Yin & Zhang (2026) observe the same pattern with drops as large as 67.5 points. The pattern extends to the strongest systems in our comparison tables: under LIBERO-Para's paraphrase protocol (Kim et al., 2026), X-VLA falls from 97.8 to 62.1 and VLA-Adapter from 98.2 to 46.3.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Table 1: **LM-free architecture and headline results.** The left panel abstracts the conditioning path: the vision branch loads a pretrained encoder, whereas the instruction branch (dashed) loads no pretrained weights and is an instruction-indexed embedding trained only by action supervision. The action head is drawn as a generic interface because the intervention is orthogonal to the action objective; Section 3 specifies the instantiation used in all reported results. The right panel summarizes SIMVLA 2.0 results; benchmark-specific protocols appear in Section 4. For BEHAVIOR-1K, we report the rank among Standard-track entries on the 2025 BEHAVIOR Challenge Public Validation leaderboard (Q-Score 0.245).



Evaluation	SIMVLA 2.0 0.3B Backbone
LIBERO Avg. (%)	96.9
LIBERO-PRO Sem. (%)	96.5
CALVIN Avg. length	4.24
WidowX Avg. (%)	91.7
Google VA Avg. (%)	82.0
RoboTwin Easy/Hard (%)	94.0/87.2
BEHAVIOR-1K	2nd (Standard)

The two findings converge. If the pretrained language component is substantially degraded by action training, and if what remains behaves like surface-form task matching on the very benchmarks used to validate VLAs, then it is not established that language pretraining is the ingredient producing instruction-conditioned control success. We do not claim this ceiling is permanent; Zhang et al. (2026a) and Yin & Zhang (2026) both propose remedies, and the mechanism is still debated, with the latter arguing that task semantics are retained internally but not routed usefully to the action head. Our point is narrower: on the task-bounded benchmarks used to validate VLAs, the assumed benefit is realized far less than the architecture implies. That makes the opposite end of the design space worth measuring. How far a policy that keeps instructions but uses no pretrained language components can go is the question this paper answers.

SIMVLA 2.0 makes that intervention explicit. **SIMVLA 2.0 uses no language model.** Throughout this paper, "language pretraining" operationally means loading (i) a pretrained language model, (ii) a pretrained text encoder, or (iii) pretrained token embeddings; SIMVLA 2.0 loads none of the three. The vision encoder's image–text pretraining is retained by default and ablated separately. We call this construction language-model-free (LM-free). It loads only a pretrained vision encoder. A parameter-free mapping assigns each instruction to deterministic fixed codes, and the corresponding embedding table is randomly initialized and updated solely by action-supervised trajectories; the mapping can be replaced by a fixed random-symbol relabeling. The resulting vectors are contextualized with visual patches and proprioception inside a joint action Transformer; they are not pretrained sentence representations.

The published SimVLA baseline controlled data order, normalization, optimization, horizon, and backbone adaptation before attributing gains to architecture on the benchmarks it studied, and retained a pretrained vision–language backbone throughout (Luo et al., 2026). We apply that controlled-recipe discipline to the component that baseline retained as pretrained—the language side of its vision–language backbone—removing pretrained language components while keeping the instruction input. The policy is still a vision-language-action model in the interface sense: it is conditioned on natural-language instructions, but it drops the pretrained language model rather than the language input.

Removing language pretraining from the action policy does not imply that language is unimportant: a full VLM remains essential for open-ended instruction understanding, task decomposition, planning, and reasoning. We view SIMVLA 2.0 as a bounded instruction-conditioned controller that can operate beneath such a high-level model, rather than as a replacement for its linguistic capabilities.

What the scores establish. Our central finding is a sufficiency result: across tabletop manipulation, controlled distribution shifts, long-horizon chains, real-to-sim evaluation on two embodiments, bimanual control, and household behavior, removing every pretrained language parameter still leaves a policy that scores in the range of recent language-pretrained systems (Table 1). Whatever language pretraining contributes to these benchmarks, it is not what puts a policy at this level. Ablations

separate the sources of performance: instruction conditioning is essential, neither pretrained token embeddings nor a full pretrained text encoder improves on action-trained codes, and strong visual pretraining remains important.

The instruction therefore carries the task signal, and LIBERO-PRO and WidowX show how far it reaches—it survives paraphrase and an unseen recombination of familiar tokens, but recomposed task goals defeat language-pretrained and LM-free policies alike.

What these benchmarks reward is instruction following that action-trained token embeddings supply as well as a pretrained language model.

Contributions.

- We construct the missing design point described above: SIMVLA 2.0 receives instructions but loads no pretrained language-side parameters and learns its instruction embeddings from action supervision alone.
- Across seven benchmark evaluations spanning six manipulation domains, SIMVLA 2.0 reaches the headline values in Table 1, placing an LM-free controller in the range of recent language-pretrained systems and second among Standard-track entries on the 2025 BEHAVIOR-1K Public Validation leaderboard.
- Controlled ablations show that instruction conditioning is essential, pretrained token embeddings provide no consistent gain, and strong visual pretraining matters. Robustness evaluations on LIBERO-PRO and WidowX further show that action-trained tokens tolerate paraphrases and unseen token recombinations.

Section 2 situates this construction among existing systems, Section 3 formalizes it, and Section 4 reports the evaluation results and ablations.

2 RELATED WORK

VLA backbones and action generation. RT-1 introduced tokenized robot control (Brohan et al., 2023b); RT-2 showed that co-fine-tuning a web-scale vision-language model yields control policies that inherit web knowledge (Brohan et al., 2023a); OpenVLA made the recipe open-source (Kim et al., 2024). How actions are decoded is a largely orthogonal axis: diffusion-based action heads descend from Diffusion Policy (Chi et al., 2024); generative and tokenized action interfaces are explored in π_0, $\pi_{0.5}$, and DD-VLA (Black et al., 2024; Physical Intelligence et al., 2025; Liang et al., 2025); and lightweight or modified decoders appear in OpenVLA-OFT, FAST, X-VLA, and VLA-Adapter (Kim et al., 2025; Pertsch et al., 2025; Zheng et al., 2025b; Wang et al., 2025a). The published SimVLA baseline controlled data order, normalization, optimization, horizon, and backbone adaptation before attributing gains to architecture (Luo et al., 2026); SIMVLA 2.0 builds on that controlled-recipe discipline while removing pretrained language components.

Efficient and no-LLM execution. Compact policies such as SmolVLA, FLOWER, and VLA-Adapter reduce model or action-head cost (Shukor et al., 2025; Reuss et al., 2025; Wang et al., 2025a). TurboVLA is a prominent efficiency-oriented system: it removes the large generative model but retains a compact pretrained text encoder (Xie et al., 2026), and VLA-0 removes large-scale action pretraining while keeping the pretrained VLM (Goyal et al., 2025). Neither removes the language-side pretraining that we study, and SIMVLA 2.0 makes no matched latency or memory claim: its variable is language-side pretraining, not parameter count or runtime.

World and video-action models. A parallel family centers pretraining on explicit future video/action prediction; recent surveys organize these systems as world action models (WAMs) or video-action (VA) models (Shen et al., 2026), with representatives such as Motus, Fast-WAM, LingBot-VA, and WorldVLA (Bi et al., 2025; Yuan et al., 2026; Li et al., 2026; Cen et al., 2025), and the family is still expanding (Zhang et al., 2026b). Such systems still typically inherit a language or vision-language backbone, so they do not answer our question; our subtraction is complementary—we keep the action interface fixed and ask whether pretrained language components are needed at all.

Task representations and language conditioning. Task IDs, learned language interfaces, and pretrained sentence representations are long-standing conditioners; classic language-conditioned

imitation learning suggests pretrained language helps synonym and held-out-task transfer (Lynch & Sermanet, 2021; Jang et al., 2022; Mees et al., 2022a). Recipe and diagnostic studies complicate this picture: the RoboVLMs study systematically varies VLA recipes (Li et al., 2024b), while diagnostic evidence shows that action fine-tuning erodes multimodal competence (Zhang et al., 2026a) and that paraphrase robustness collapses across architectures and scales (Kim et al., 2026; Yin & Zhang, 2026). LIBERO-PRO complements these diagnostics with controlled object, position, semantic-instruction, and task perturbations of the standard LIBERO suites (Zhou et al., 2025). These lines motivate our narrower question—whether fixed symbols suffice for task-bounded control—and differentiate us from recipe exploration: SIMVLA 2.0 removes pretrained language-side parameters, while our ablations separately isolate instruction conditioning, pretrained token embeddings, and visual pretraining.

Cross-domain control. Generalist policies such as Octo and Open X-Embodiment target transfer across datasets and embodiments (Octo Model Team et al., 2024; Open X-Embodiment Collaboration et al., 2024), while the protocols here span real-to-sim manipulation (SimplerEnv (Li et al., 2024c)), bimanual control (RoboTwin 2.0 (Chen et al., 2025)), and long-horizon household activity (BEHAVIOR-1K (Li et al., 2024a)). We evaluate across these complementary settings under their respective standard protocols.

3 SIMVLA 2.0

3.1 PROBLEM FORMULATION

At control step t, the policy receives valid RGB views $I_t^{1:V}$, proprioceptive state s_t, and instruction ℓ, and predicts

$$A_t = [a_t, \ldots, a_{t+H-1}] \in \mathbb{R}^{H \times d_a}.$$

At most $h \leq H$ actions are executed before replanning. Each benchmark uses its corresponding action adapter, normalization, H, h, and evaluator. The shared goal is to model $\pi(A_t \mid I_t^{1:V}, s_t, \ell)$ without any pretrained language model, text encoder, or token embedding.

3.2 VISION-ONLY ENCODER AND LM-FREE INSTRUCTION CODES

Vision encoder. Each view is encoded independently by a vision encoder E_ϕ^{vis},

$$Z_t^v = E_\phi^{\text{vis}}(I_t^{1:V}),$$

which returns patch tokens. We state this component abstractly: the construction requires only a vision-only backbone carrying no language-model component. The intervention studied here concerns language pretraining rather than any particular visual backbone.

The reported runs instantiate E_ϕ^{vis} with pretrained SigLIP2-L/16 vision weights (Tschannen et al., 2025), retaining patch tokens and discarding the classification head. We instantiate only the vision tower of SigLIP2 and never build its text tower, so SigLIP2 supplies visual pretraining without a language-model component.

Fixed instruction codes. A fixed tokenizer q, with no trainable parameters, maps ℓ to a sequence of discrete token IDs $q(\ell) = (w_1, \ldots, w_L)$, where each w_i represents a word or WordPiece subword. The corresponding instruction tokens are

$$Z^\ell = [W_{w_1}, \ldots, W_{w_L}], \qquad W \in \mathbb{R}^{|\mathcal{V}| \times d},$$

where W is randomly initialized and updated exclusively by the action-supervised objective. No text-only objective, pretrained token geometry, or dedicated contextual text encoder is used. In the reported instantiation the lookup width d equals the vision encoder's output width (1024 for the SigLIP2-L runs), so no separate text projection is added at the embedding level. A fixed random-symbol relabeling may permute code identities before training; because both the mapping and permutation are fixed, neither introduces learned linguistic information. Codes preserve discrete identity, sequence order, and shared symbols only; shared symbols across instructions are the only way different instructions share token geometry. Thus SIMVLA 2.0 removes language pretraining, not instruction input or subsequent multimodal contextual computation.

3.3 JOINT ACTION TRANSFORMER

The trunk is orthogonal to the action objective; we describe the flow-matching instantiation used by all reported runs, whose auxiliary input is a noise level τ and a corrupted chunk x_τ. Each noisy action token $x_{\tau,i}$, the proprioceptive state s_t, and a sinusoidal time embedding $e(\tau)$ are concatenated and mapped by a learned linear projection P_a to the trunk width d_{model}:

$$h_i^a = P_a[x_{\tau,i}; s_t; e(\tau)].$$

Vision and instruction tokens are mapped by one shared learned linear projection P_{vlm}, so the encoder input is

$$H_0 = [h_1^a, \ldots, h_H^a; P_{\text{vlm}} Z_t^v; P_{\text{vlm}} Z^\ell],$$

A learned positional embedding is added, and the sequence is processed by a standard bidirectional Transformer encoder (Vaswani et al., 2017). A linear decoder on the first H positions predicts an action-space velocity. In the reported runs, P_a maps the concatenation of action, proprioceptive, and time-embedding inputs to d_{model}, while P_{vlm} maps the shared vision/lookup width d (1024) to the trunk width d_{model} (1024 in all six training domains, Table 12). Because instruction tokens attend jointly to vision, noisy actions, and τ, their hidden states are learned under action supervision and conditioned on noisy actions and flow time, rather than being reusable sentence embeddings.

3.4 ACTION HEAD

The components above produce a conditioning context

$$c_t = g_\theta\big(I_t^{1:V}, s_t, \ell\big)\,,$$

from which an action head $\mathcal{H}_\theta$ decodes the chunk A_t. We state the head abstractly because the intervention studied in this paper acts on how c_t is formed, not on how A_t is decoded from it. Let z denote whatever auxiliary input a head consumes, y the matching supervision target, and ℓ_{obj} the per-sample objective. Training then minimizes

$$\mathcal{L}(\theta) = \mathbb{E}_{(c_t, A_t)\sim\mathcal{D},\, z\sim p(z|A_t)}\Big[\ell_{\text{obj}}\big(\mathcal{H}_\theta(c_t, z),\, y(A_t, z)\big)\Big],$$

and the usual action objectives are recovered by choosing $(z, y, \ell_{\text{obj}})$:

- **Direct regression.** $z = \varnothing$ and $y = A_t$, with ℓ_{obj} a squared or Huber error; the chunk is emitted in one pass.
- **Iterative denoising.** $z = (\tau, x_\tau)$ pairs a noise level with a corrupted chunk and y is the corresponding velocity or noise, with ℓ_{obj} a squared error. Flow matching and diffusion differ only in the corruption schedule and the target parameterization.
- **Autoregressive decoding.** z is a prefix of already-emitted tokens of a discretized chunk, y is the next token, and ℓ_{obj} is a cross-entropy.

All results reported here use a conditional flow-matching instantiation (Lipman et al., 2022), written out in Section A.4; we therefore make no empirical claim about the remaining objectives. Actions and proprioception are normalized per dimension with benchmark training statistics and transformed back to native controls before execution. Benchmark-specific optimization, hyperparameter, and control-interface details are provided in Tables 11 and 12.

4 EVALUATION

4.1 EXPERIMENTAL SETUP

Benchmarks. **LIBERO** contains four ten-task suites and reports suite and macro success (Liu et al., 2023). **LIBERO-PRO** augments these suites with controlled changes to object attributes, initial positions, semantically equivalent instructions, and task goals (Zhou et al., 2025). **CALVIN** ABC→D evaluates 1,000 chains of five subtasks in held-out scene D and reports SR@1–5 and average completed chain length (Mees et al., 2022b). **SimplerEnv** evaluates **BridgeData V2** training on four **WidowX** tasks and Fractal training on Google Robot variant aggregations (Li et al., 2024c; Walke et al., 2023; Brohan et al., 2023b). **RoboTwin 2.0** evaluates bimanual control under the standard Easy/Hard (clean/randomized) settings (Chen et al., 2025). **BEHAVIOR-1K** covers long-horizon household activities (Li et al., 2024a); we report the challenge's official Q-Score (computed internally as the per-task mean over rollouts, `q_mean`).

Table 2: **BEHAVIOR-1K leaderboard comparison.** Q-Score on the 2025 BEHAVIOR Challenge Public Validation leaderboard; SIMVLA 2.0 is highlighted in bold. All rows are Standard track except Embodied Intelligence (†Privileged).

Rank	Team	Q-Score (Public Val.)
1	Robot Learning Collective (Robot Learning Collective, 2025)	0.2605
2	**SIMVLA 2.0 (Ours)**	**0.2450**
3	SimpleAI Robot (Beijing Simple AI Technology Co., Ltd., 2025)	0.1943
4	Comet (NVIDIA Research, 2025)	0.1830
5	The North Star (Huawei CRI EAI Team, 2025)	0.1702
6	Embodied Intelligence† (Embodied Intelligence, 2025)	0.1110

We follow the standard protocol for each of the seven benchmark evaluations, which span six task domains and their corresponding control interfaces. Complete protocol, optimization, hyperparameter, and control-interface details are provided in Tables 10 to 12; baseline values follow the corresponding cited sources. In the results below we lead with BEHAVIOR-1K, the most demanding setting in this suite, and then follow the benchmark order above.

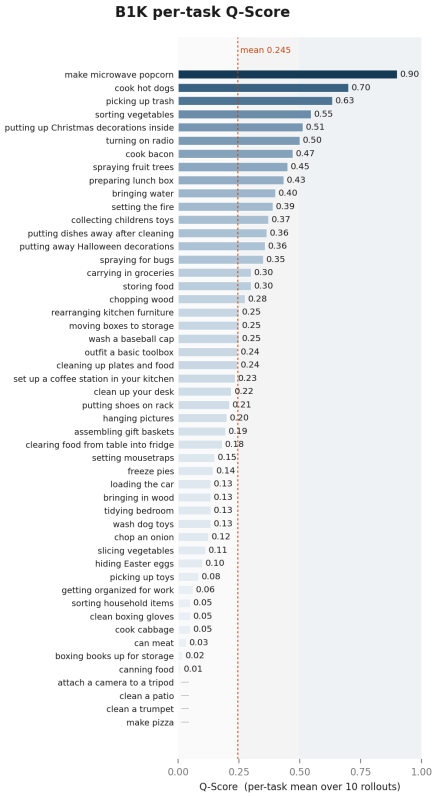


Figure 1: **BEHAVIOR-1K SIMVLA 2.0 per-task results.** Per-task Q-Score over 10 rollouts for all evaluated household behaviors (dashed line: mean 0.245). Q-Score is the challenge's official partial-credit metric; internally we compute it as `q_mean`, the per-task mean over rollouts.

4.2 HEADLINE RESULTS

BEHAVIOR-1K: Long-Horizon Household Behavior. BEHAVIOR-1K is the most demanding setting in our suite: the 2025 BEHAVIOR Challenge scores 50 full-length household activities in OmniGibson and supplies 10,000 teleoperated expert demonstrations for training (Li et al., 2024a). Those demonstrations average 397 seconds and carry 27.06 annotated skill segments per trajectory drawn from 31 unique skills. Figure 1 reports per-task Q-Score for all 50 tasks; SIMVLA 2.0 places second among Standard-track entries with a leaderboard Q-Score of 0.245 (0.2605 for the top entry, 0.1943 for the third), using a policy that loads no pretrained language parameters (Table 2). Q-Score is the challenge's official partial-credit metric over satisfied BDDL goal predicates, computed internally as `q_mean`, the per-task mean over rollouts.

Its expert demonstrations average 6.6 minutes, while the longest activities extend to 14.4 minutes and approximately 25,900 visual frames. Despite these horizons, SIMVLA 2.0 obtains non-zero Q-Score on eight of the nine activities whose expert demonstrations exceed nine minutes. Two long multi-stage activities outperform the overall 50-task mean: *storing food* reaches 0.30 over an 11.1-minute demonstration horizon (approximately 20,000 frames), and *collecting children's toys* reaches 0.37 over 10.6 minutes (approximately 19,000 frames). The former requires a multi-room sequence—retrieving groceries from a car trunk in the garage, carrying them to the kitchen, placing items in the refrigerator, and closing both the trunk and refrigerator. Together with the second-place Standard-track result, these outcomes show that SIMVLA 2.0 can sustain meaningful progress through long, multi-stage household activities using onboard observations, without a pretrained language model, explicit subgoal decomposition, or test-time search. Figure 2 in Section A.6 shows goal-satisfying episodes on eight further activities, including *cook bacon* and *setting the fire*.

LIBERO and CALVIN. On LIBERO, SIMVLA 2.0 achieves 96.9% average success across the four task suites, including 92.2% on LIBERO-Long, placing the LM-free policy among the strongest recent VLA systems (Table 3). On CALVIN ABC→D, it reaches an average completed chain

Table 3: **LIBERO comparison.** Success (%). "B" is the scale of the *pretrained* backbone in billions.

Model	B	Spatial	Object	Goal	Long	Avg
Large Models (≥ 4B)						
UniVLA (Bu et al., 2025)	9	96.5	96.8	95.6	92.0	95.2
FlowVLA (Zhong et al., 2025)	8.5	93.2	95.0	91.6	72.6	88.1
UnifiedVLA (Wang et al., 2025b)	8.5	95.4	98.8	93.6	94.0	95.5
OpenVLA (Kim et al., 2024)	7	84.7	88.4	79.2	53.7	76.5
OpenVLA-OFT (Kim et al., 2025)	7	97.6	98.4	97.9	94.5	97.1
DD-VLA (Liang et al., 2025)	7	97.2	98.6	97.4	92.0	96.3
MemoryVLA (Shi et al., 2025)	7	98.4	98.4	96.4	93.4	96.7
PD-VLA (Song et al., 2025)	7	95.5	96.7	94.9	91.7	94.7
MolmoAct (Lee et al., 2025)	7	87.0	95.4	87.6	77.2	86.8
ThinkAct (Huang et al., 2025)	7	88.3	91.4	87.1	70.9	84.4
CoT-VLA (Zhao et al., 2025)	7	87.5	91.6	87.6	69.0	83.9
WorldVLA (Cen et al., 2025)	7	87.6	96.2	83.4	60.0	81.8
TraceVLA (Zheng et al., 2024)	7	84.6	85.2	75.1	54.1	74.8
FPC-VLA (Yang et al., 2025)	7	86.2	87.0	92.0	82.2	86.9
4D-VLA (Zhang et al., 2025)	4	88.9	95.2	90.9	79.1	88.5
SpatialVLA (Qu et al., 2025)	4	88.2	89.9	78.6	55.5	78.1
Small Models (< 4B)						
π_0 (Black et al., 2024)	3	96.8	98.8	95.8	85.2	94.2
π_0-FAST (Pertsch et al., 2025)	3	96.4	96.8	88.6	60.2	85.5
$\pi_{0.5}$ (Physical Intelligence et al., 2025)	3	98.8	98.2	98.0	92.4	96.9
NORA (Hung et al., 2025)	3	92.2	95.4	89.4	74.6	87.9
SmolVLA (Shukor et al., 2025)	2.2	93.0	94.0	91.0	77.0	88.8
VLANeXt (Wu et al., 2026b)	2	99.0	99.2	96.6	94.6	97.4
GR00T-N1 (NVIDIA et al., 2025)	2	94.4	97.6	93.0	90.6	93.9
GraspVLA (Deng et al., 2025)	1.8	-	94.1	91.2	82.0	89.1
FLOWER (Reuss et al., 2025)	1	97.5	99.1	96.1	94.9	96.9
Tiny Models (< 1B)						
X-VLA (Zheng et al., 2025b)	0.9	98.2	98.6	97.8	97.6	98.1
VLA-Adapter (Wang et al., 2025a)	0.5	97.8	99.2	97.2	95.0	97.3
VLA-OS (Gao et al., 2025)	0.5	87.0	96.5	92.7	66.0	85.6
UniAct (Zheng et al., 2025a)	0.5	77.0	87.0	77.0	70.0	77.8
SimVLA (Luo et al., 2026)	0.5	99.6	99.8	98.6	96.4	98.6
SIMVLA 2.0	0.3	98.4	99.6	97.4	92.2	96.9

Table 4: **CALVIN ABC→D comparison.** Success (%) and average chain length.

Model	SR@1	SR@2	SR@3	SR@4	SR@5	Avg. len
Large (≥ 4B)						
UniVLA (Bu et al., 2025)	95.5	85.8	75.4	66.9	56.5	3.80
UnifiedVLA (Wang et al., 2025b)	98.9	94.8	89.0	82.8	75.1	4.41
OpenVLA (Kim et al., 2024)	91.3	77.8	62.0	52.1	43.5	3.27
OpenVLA-OFT (Kim et al., 2025)	96.3	89.1	82.4	75.8	66.5	4.10
HiF-VLA (Lin et al., 2025)	98.5	94.1	88.1	81.4	73.1	4.35
Tiny (< 1B)						
Seer (Tian et al., 2024)	96.3	91.6	86.1	80.3	74.0	4.28
VLA-Adapter (Wang et al., 2025a)	99.1	94.6	88.8	82.8	76.5	4.42
X-VLA (Zheng et al., 2025b)	97.1	92.6	88.5	84.4	78.8	4.43
SimVLA (Luo et al., 2026)	99.4	96.4	89.8	83.7	77.2	4.47
SIMVLA 2.0	99.4	93.0	85.8	78.6	67.6	4.24

Table 5: **RoboTwin 2.0 comparison.** Success (%) under the standard Easy/Hard settings (`eval_clean`/`eval_randomized`).

Model	Easy	Hard	Avg
π_0 (Black et al., 2024)	65.9	58.4	62.2
$\pi_{0.5}$ (Physical Intelligence et al., 2025)	82.7	76.8	79.8
LingBot-VLA (Wu et al., 2026a)	86.5	85.3	85.9
Motus (Bi et al., 2025)	88.7	87.0	87.8
StarVLA (StarVLA Community, 2026)	88.7	87.8	88.3
Fast-WAM (Yuan et al., 2026)	91.9	91.8	91.8
LingBot-VA (Li et al., 2026)	92.9	91.5	92.2
G0.5 (Liu et al., 2026)	93.7	92.8	93.3
SIMVLA 2.0	94.0	87.2	90.6

Table 6: **WidowX comparison.** Success (%).

Method	Spoon	Carrot	Stack	Eggplant	Avg
Large (≥ 4B)					
RT-1-X (Open X-Embodiment Collaboration et al., 2024)	0.0	4.2	0.0	0.0	1.1
Octo-Base (Octo Model Team et al., 2024)	12.5	8.3	0.0	43.1	16.0
OpenVLA (Kim et al., 2024)	0.0	0.0	0.0	4.1	1.0
OpenVLA-OFT (Kim et al., 2025)	12.5	4.2	8.3	37.5	15.6
DD-VLA (Liang et al., 2025)	29.2	29.2	20.8	70.8	37.5
RoboVLM (Li et al., 2024b)	29.2	25.0	12.5	58.3	31.3
SpatialVLA (Qu et al., 2025)	16.7	25.0	29.2	100.0	42.7
MemoryVLA (Shi et al., 2025)	75.0	75.0	37.5	100.0	71.9
ThinkAct (Huang et al., 2025)	58.3	37.5	8.7	70.8	43.8
FPC-VLA (Yang et al., 2025)	79.2	58.3	45.8	75.0	64.6
Small (< 4B)					
Octo-Small (Octo Model Team et al., 2024)	47.2	9.7	4.2	56.9	29.5
π_0 (Black et al., 2024)	29.1	0.0	16.7	62.5	27.1
π_0-FAST (Pertsch et al., 2025)	29.1	21.9	10.8	66.6	32.1
GR00T-N1 (NVIDIA et al., 2025)	62.5	45.8	16.7	20.8	36.5
FLOWER (Reuss et al., 2025)	71.0	13.0	8.0	88.0	45.0
Tiny (< 1B)					
X-VLA (Zheng et al., 2025b)	100.0	91.7	95.8	95.8	95.8
SimVLA (Luo et al., 2026)	100.0	100.0	91.7	91.7	95.8
SIMVLA 2.0	100.0	100.0	79.2	87.5	91.7

Table 7: **Google Robot comparison.** Success (%).

Model	Pick	Move	Open	Avg
Large (≥ 4B)				
RT-1-X (Open X-Embodiment Collaboration et al., 2024)	49.0	32.3	29.4	36.9
RT-2-X (Open X-Embodiment Collaboration et al., 2024)	82.3	79.2	35.3	65.6
Octo-Base (Octo Model Team et al., 2024)	0.6	3.1	1.1	1.6
OpenVLA (Kim et al., 2024)	54.5	47.7	17.7	40.0
OpenVLA-OFT (Kim et al., 2025)	65.3	59.0	12.2	45.5
RoboVLM (Li et al., 2024b)	75.6	60.0	10.6	48.7
TraceVLA (Zheng et al., 2024)	60.0	56.4	31.0	49.1
DD-VLA (Liang et al., 2025)	82.5	64.6	23.6	56.9
SpatialVLA (Qu et al., 2025)	88.0	72.7	41.8	67.5
ThinkAct (Huang et al., 2025)	84.0	63.8	47.6	65.1
Small (< 4B)				
π_0 (Black et al., 2024)	75.2	63.7	25.6	54.8
π_0-FAST (Pertsch et al., 2025)	77.6	68.2	31.3	59.0
GR00T-N1 (NVIDIA et al., 2025)	78.8	62.5	13.2	51.5
Tiny (< 1B)				
X-VLA (Zheng et al., 2025b)	85.5	79.8	61.9	75.7
SimVLA (Luo et al., 2026)	82.3	81.0	67.7	77.0
SIMVLA 2.0	89.7	76.5	79.9	82.0

length of 4.24 out of five in the held-out scene, demonstrating sustained multi-step execution under instruction-conditioned control (Table 4). Together, these results show that pretrained language-side parameters are not required for high success across either diverse manipulation suites or compositional multi-step sequences.

LIBERO-PRO. SIMVLA 2.0 retains nonzero mean success under every evaluated perturbation type (Table 8). Most notably, it achieves 96.5% under semantic instruction perturbations, closely matching its 96.9% original-task average despite loading no pretrained language parameters. Recomposed task goals, which the scene cannot disambiguate, defeat every method: $\pi_{0.5}$ 0.8%, SimVLA 6.0%, and SIMVLA 2.0 6.5%. Pretrained language-side parameters therefore buy no compositional goal grounding at the point where it would be required. The simplicity of our construction is what makes this legible: with no pretrained language to credit, instruction following on these suites is supplied by randomly initialized, action-trained token embeddings as well as by a pretrained language model.

RoboTwin 2.0. SIMVLA 2.0 achieves 94.0% success in the Easy setting and 87.2% under randomized Hard evaluation, yielding a 90.6% average across the two settings (Table 5). The Easy result is the highest in the comparison, while the Hard result remains competitive with leading vision-language-action and video-action policies. This performance extends the LM-free construction from single-arm manipulation to coordinated bimanual control under visual and object-level variation.

WidowX and Google Robot. SIMVLA 2.0 achieves 91.7% average success on WidowX, including perfect success on Spoon on Towel and Carrot on Plate, and ranks third among the compared methods (Table 6). At evaluation time, the SimplerEnv client forwards the environment-provided instruction directly to the policy; during Bridge training, raw annotations are cleaned and stochastically replaced by augmentation variants. SIMVLA 2.0 remains effective under this prompt shift, reaching 79.2% on Stack Cube and 87.5% on Eggplant in Basket; the latter evaluation prompt appears neither verbatim nor as an augmentation variant in the training instructions, providing evidence of robustness to prompt variation and seen-token recombination. Together with the explicit semantic perturbations in LIBERO-PRO (Table 8), WidowX provides complementary evidence that LM-free instruction conditioning can tolerate both controlled paraphrases and naturally shifted evaluation prompts. On Google Robot, it reaches an 82.0% variant-aggregation average, the strongest result among all methods in Table 7. These evaluations extend the LM-free result across distinct robot embodiments, visual domains, and controller semantics in SimplerEnv.

Comparison against the language-pretrained construction. Among the tabulated systems, SimVLA is the closest matched reference point: it differs principally in retaining the pretrained language side of a vision–language backbone (Luo et al., 2026). The two constructions therefore bracket what pretrained language parameters contribute under an otherwise comparable setup. Across the shared evaluations the margin is small and does not favor one side uniformly: SIMVLA 2.0 trails by 1.7 points in LIBERO average success, by 0.23 in CALVIN average chain length, while exceeding the language-pretrained configuration by 5.0 points on Google Robot. Under LIBERO-PRO the two remain within 2.3 points on semantic instruction perturbations (96.5% versus 98.8%), whereas SIMVLA 2.0 is substantially stronger under initial-position perturbation (21.5% versus 8.3%). These results are obtained with a 0.3B vision backbone in place of a 0.5B vision–language backbone and with no loaded language parameters. We therefore read the comparison as evidence that pretrained language parameters yield a modest and benchmark-dependent margin on in-distribution manipulation suites, rather than a capability that instruction-conditioned control requires.

4.3 ABLATIONS

This section identifies which components of the construction are responsible for the reported performance (Table 9). The no-instruction variant removes the instruction pathway. We additionally compare the default random, action-trained instruction embeddings with pretrained language-side alternatives (token embedding table and full text encoder) derived from BERT (Devlin et al., 2019) and SigLIP2 (Tschannen et al., 2025). On the vision side, the DINOv3 variant replaces SigLIP2 with a vision encoder pretrained without text supervision (Siméoni et al., 2025), and the no-vision-pretraining variant trains the same SigLIP2 encoder architecture from random initialization. Table 9 reports all component scores and their mean in percent.

Instruction conditioning provides the task signal. Removing instruction conditioning reduces LIBERO average success from 96.9% to 47.6% and WidowX average success from 91.7% to 59.4%. On LIBERO, the largest reductions occur on Goal and Long, whereas Object remains at 93.4%, indicating that visual observations can identify some object-centric behaviors but are insufficient when the same scene admits distinct goals or extended task sequences. The result directly separates the role of instruction *conditioning* from language *pretraining*: pretrained language parameters are unnecessary, but the action-trained instruction codes carry essential task information.

Pretrained text representations provide no advantage. We test two levels of language-side pretraining while holding the vision encoder, action Transformer, data, and optimization fixed. The embedding-table variants replace the random lookup table with BERT's or SigLIP2's pretrained token embeddings, which remain trainable under the action objective; the text-encoder variant instead feeds the contextual token outputs of the full pretrained SigLIP2 text tower to the action Transformer. The

Table 8: **Robustness evaluation on LIBERO-PRO.** Success rate (%) on the original tasks (Ori) and under object-attribute (Obj), initial-position (Pos), semantic-instruction (Sem), and task-goal (Task) perturbations (Zhou et al., 2025).

Suite	Method	Ori	Obj	Pos	Sem	Task
Spatial	OpenVLA (Kim et al., 2024)	98.0	97.0	0.0	97.0	0.0
	π_0 (Black et al., 2024)	96.8	95.0	0.0	97.0	0.0
	$\pi_{0.5}$ (Physical Intelligence et al., 2025)	98.0	97.0	20.0	97.0	1.0
	SimVLA (Luo et al., 2026)	99.0	98.0	29.0	98.0	0.0
	SIMVLA 2.0	98.4	98.0	40.0	96.0	0.0
Object	OpenVLA (Kim et al., 2024)	99.0	98.0	0.0	98.0	0.0
	π_0 (Black et al., 2024)	98.8	94.0	0.0	90.0	0.0
	$\pi_{0.5}$ (Physical Intelligence et al., 2025)	98.0	98.0	17.0	96.0	1.0
	SimVLA (Luo et al., 2026)	100.0	85.0	1.0	100.0	4.0
	SIMVLA 2.0	99.6	90.0	28.0	100.0	10.0
Goal	OpenVLA (Kim et al., 2024)	98.0	96.0	0.0	98.0	0.0
	π_0 (Black et al., 2024)	95.8	94.0	0.0	93.0	0.0
	$\pi_{0.5}$ (Physical Intelligence et al., 2025)	97.0	97.0	38.0	97.0	0.0
	SimVLA (Luo et al., 2026)	99.0	82.0	0.0	99.0	10.0
	SIMVLA 2.0	97.4	78.0	4.0	96.0	10.0
Long	OpenVLA (Kim et al., 2024)	93.0	81.0	0.0	96.0	0.0
	π_0 (Black et al., 2024)	85.2	79.0	0.0	82.0	0.0
	$\pi_{0.5}$ (Physical Intelligence et al., 2025)	93.0	92.0	8.0	93.0	1.0
	SimVLA (Luo et al., 2026)	96.0	61.0	3.0	98.0	10.0
	SIMVLA 2.0	92.2	54.0	14.0	94.0	6.0

Table 9: **Ablations on LIBERO and WidowX (%).** The reported SIMVLA 2.0 result is the default configuration; each ablation changes one component while retaining the remaining architecture and training recipe.

(a) LIBERO

Knob	Value	Spatial	Object	Goal	Long	Avg.
SIMVLA 2.0	**Default**	**98.4**	**99.6**	**97.4**	**92.2**	**96.9**
Instruction conditioning	None	75.4	93.4	6.4	15.2	47.6
Text embedding	BERT emb. table	99.8	99.0	97.6	90.8	96.8
	SigLIP2 emb. table	98.8	99.2	96.4	84.4	94.7
	SigLIP2 text encoder	96.2	98.8	88.2	82.0	91.3
Vision backbone	DINOv3	95.6	95.0	89.4	80.2	90.1
	No vision pretraining	69.2	85.0	66.8	16.2	59.3

(b) WidowX

Knob	Value	Carrot	Eggplant	Spoon	Stack	Avg.
SIMVLA 2.0	**Default**	**100.0**	**87.5**	**100.0**	**79.2**	**91.7**
Instruction conditioning	None	83.3	20.8	91.7	41.7	59.4
Text embedding	BERT emb. table	100.0	91.7	95.8	75.0	90.6
	SigLIP2 emb. table	95.8	50.0	100.0	83.3	82.3
	SigLIP2 text encoder	95.8	41.7	100.0	83.3	80.2
Vision backbone	DINOv3	91.7	0.0	87.5	8.3	46.9
	No vision pretraining	16.7	4.2	25.0	0.0	11.5

BERT table reaches 96.8% on LIBERO and 90.6% on WidowX; the SigLIP2 table reaches 94.7% and 82.3%, and the full SigLIP2 text encoder 91.3% and 80.2%. Neither pretrained token geometry nor a contextual pretrained text encoder improves on the random, action-trained embeddings.

Visual pretraining matters. Replacing SigLIP2 with DINOv3 retains 90.1% average success on LIBERO, although the WidowX average falls to 46.9%, suggesting that WidowX may benefit more strongly from language-aligned visual pretraining. Randomly initializing and training the same SigLIP2 vision architecture from scratch reduces performance further to 59.3% on LIBERO and 11.5% on WidowX. Thus, strong visual pretraining is important.

Seen-symbol recombination. The WidowX instruction "put eggplant into yellow basket" appears neither verbatim nor as an augmentation variant in the 19,376 distinct Bridge training instructions, although all of its tokens are individually covered by the training vocabulary. SIMVLA 2.0 reaches 87.5% on this task, compared with 20.8% for the no-instruction variant. Combined with 96.5% mean success under LIBERO-PRO semantic instruction perturbations, this result is consistent with composition over action-trained token embeddings rather than whole-instruction memorization, and further supports that the instruction pathway contributes task-relevant information beyond the visual observation alone.

5 CONCLUSION

SIMVLA 2.0 keeps instruction conditioning while removing all pretrained language-side parameters. Across seven evaluations spanning six manipulation domains, it remains competitive with language-pretrained systems and places second on the BEHAVIOR-1K Standard track. The ablations show that instructions remain essential, pretrained text representations provide no consistent advantage, and visual pretraining still matters. Thus, these task-bounded control benchmarks do not require pretrained language-side parameters.

Language remains the brain. Language remains central to open-ended instruction understanding, task decomposition, planning, reasoning, and interaction. We therefore advocate a modular brain–cerebellum organization: a full VLM handles high-level semantics and reasoning, while SIMVLA 2.0 executes bounded skills as a instruction-conditioned controller.

Limitations. Our two 8×H100 nodes did not permit per-domain hyperparameter search; BEHAVIOR-1K in particular uses a single heuristically chosen configuration, so further tuning may improve its Q-Score. We evaluate only the instruction-conditioned controller; integrating it with a high-level VLM planner remains future work.

AI USE STATEMENT

Generative AI tools assisted manuscript restructuring and language editing. They were not used to generate experimental measurements. The authors independently reviewed the experimental results, implementation, claims, and citations and remain responsible for the final paper and artifacts.

REFERENCES

Beijing Simple AI Technology Co., Ltd. SimpleAI Robot: 2025 BEHAVIOR Challenge Public Validation entry (rank 3, Standard track). 2025 BEHAVIOR Challenge public leaderboard, 2025.

Hongzhe Bi, Hengkai Tan, Shenghao Xie, Zeyuan Wang, Shuhe Huang, Haitian Liu, Ruowen Zhao, Yao Feng, Chendong Xiang, Yinze Rong, Hongyan Zhao, Hanyu Liu, Zhizhong Su, Lei Ma, Hang Su, and Jun Zhu. Motus: A Unified Latent Action World Model. *arXiv preprint arXiv:2512.13030*, 2025. URL https://arxiv.org/abs/2512.13030.

Kevin Black, Noah Brown, Danny Driess, Chelsea Finn, Karol Hausman, Sergey Levine, et al. π_0: A Vision-Language-Action Flow Model for General Robot Control. *arXiv preprint arXiv:2410.24164*, 2024. URL https://arxiv.org/abs/2410.24164.

Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Choromanski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, Pete Florence, Chuyuan Fu, Montse Gonzalez Arenas, Keerthana Gopalakrishnan, Kehang Han, Karol Hausman, Alexander Herzog, Jasmine Hsu, Brian Ichter, Alex Irpan, Nikhil Joshi, Ryan Julian, Dmitry Kalashnikov, Yuheng Kuang, Isabel Leal, Lisa Lee, Tsang-Wei Edward Lee, Sergey Levine, Yao Lu, Henryk Michalewski, Igor Mordatch, Karl Pertsch, Kanishka Rao, Krista Reymann, Michael Ryoo, Grecia Salazar, Pannag Sanketi, Pierre Sermanet, Jaspiar Singh, Anikait Singh, Radu Soricut, Huong Tran, Vincent Vanhoucke, Quan Vuong, Ayzaan Wahid, Stefan Welker, Paul Wohlhart, Jialin Wu, Fei Xia, Ted Xiao, Peng Xu, Sichun Xu, Tianhe Yu, and Brianna Zitkovich. RT-2: Vision-Language-Action Models Transfer Web Knowledge to Robotic Control. *arXiv preprint arXiv:2307.15818*, 2023a. URL https://arxiv.org/abs/2307.15818.

Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Joseph Dabis, Chelsea Finn, Keerthana Gopalakrishnan, Karol Hausman, Alexander Herzog, Jasmine Hsu, et al. RT-1: Robotics Transformer for Real-World Control at Scale. In *Proceedings of Robotics: Science and Systems*, Daegu, Republic of Korea, July 2023b. doi: 10.15607/RSS.2023.XIX.025. URL https://arxiv.org/abs/2212.06817.

Qingwen Bu, Yanting Yang, Jisong Cai, Shenyuan Gao, Guanghui Ren, Maoqing Yao, Ping Luo, and Hongyang Li. UniVLA: Learning to Act Anywhere with Task-centric Latent Actions. *arXiv preprint arXiv:2505.06111*, 2025. URL https://arxiv.org/abs/2505.06111.

Jun Cen, Chaohui Yu, Hangjie Yuan, Yuming Jiang, Siteng Huang, Jiayan Guo, Xin Li, Yibing Song, Hao Luo, Fan Wang, Deli Zhao, and Hao Chen. WorldVLA: Towards Autoregressive Action World Model. *arXiv preprint arXiv:2506.21539*, 2025. URL https://arxiv.org/abs/2506.21539.

Tianxing Chen, Zanxin Chen, Baijun Chen, Zijian Cai, Yibin Liu, Zixuan Li, Qiwei Liang, Xianliang Lin, Yiheng Ge, Zhenyu Gu, Weiliang Deng, Yubin Guo, Tian Nian, Xuanbing Xie, Qiangyu Chen, Kailun Su, Tianling Xu, Guodong Liu, Mengkang Hu, Huan-ang Gao, Kaixuan Wang, Zhixuan Liang, Yusen Qin, Xiaokang Yang, Ping Luo, and Yao Mu. RoboTwin 2.0: A Scalable Data Generator and Benchmark with Strong Domain Randomization for Robust Bimanual Robotic Manipulation. *arXiv preprint arXiv:2506.18088*, 2025. URL https://arxiv.org/abs/2506.18088.

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, 2024. doi: 10.1177/02783649241273668. URL https://arxiv.org/abs/2303.04137.

Shengliang Deng, Mi Yan, Songlin Wei, Haixin Ma, Yuxin Yang, Jiayi Chen, Zhiqi Zhang, Taoyu Yang, Xuheng Zhang, Wenhao Zhang, Heming Cui, Zhizheng Zhang, and He Wang. GraspVLA: a Grasping Foundation Model Pre-trained on Billion-scale Synthetic Action Data. *arXiv preprint arXiv:2505.03233*, 2025. URL https://arxiv.org/abs/2505.03233.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv preprint arXiv:1810.04805*, 2019. URL https://arxiv.org/abs/1810.04805.

Embodied Intelligence. Embodied Intelligence: 2025 BEHAVIOR Challenge Public Validation entry (rank 6, Privileged track). 2025 BEHAVIOR Challenge public leaderboard. Affiliation: Independent, 2025.

Chongkai Gao, Zixuan Liu, Zhenghao Chi, Junshan Huang, Xin Fei, Yiwen Hou, Yuxuan Zhang, Yudi Lin, Zhirui Fang, Zeyu Jiang, and Lin Shao. VLA-OS: Structuring and Dissecting Planning Representations and Paradigms in Vision-Language-Action Models. *arXiv preprint arXiv:2506.17561*, 2025. URL https://arxiv.org/abs/2506.17561.

Ankit Goyal, Hugo Hadfield, Xuning Yang, Valts Blukis, and Fabio Ramos. VLA-0: Building State-of-the-Art VLAs with Zero Modification. *arXiv preprint arXiv:2510.13054*, 2025. URL https://arxiv.org/abs/2510.13054.

Chi-Pin Huang, Yueh-Hua Wu, Min-Hung Chen, Yu-Chiang Frank Wang, and Fu-En Yang. ThinkAct: Vision-Language-Action Reasoning via Reinforced Visual Latent Planning. *arXiv preprint arXiv:2507.16815*, 2025. URL https://arxiv.org/abs/2507.16815.

Huawei CRI EAI Team. The North Star: 2025 BEHAVIOR Challenge Public Validation entry (rank 5, Standard track). 2025 BEHAVIOR Challenge public leaderboard, 2025.

Chia-Yu Hung, Qi Sun, Pengfei Hong, Amir Zadeh, Chuan Li, U-Xuan Tan, Navonil Majumder, and Soujanya Poria. NORA: A Small Open-Sourced Generalist Vision Language Action Model for Embodied Tasks. *arXiv preprint arXiv:2504.19854*, 2025. URL https://arxiv.org/abs/2504.19854.

Eric Jang, Alex Irpan, Mohi Khansari, Daniel Kappler, Frederik Ebert, Corey Lynch, Sergey Levine, and Chelsea Finn. BC-z: Zero-shot task generalization with robotic imitation learning. In *Proceedings of the 5th Conference on Robot Learning*, volume 164 of *Proceedings of Machine Learning Research*, pp. 991–1002. PMLR, 2022. URL https://proceedings.mlr.press/v164/jang22a.html.

Chanyoung Kim, Minwoo Kim, Minseok Kang, Hyunwoo Kim, and Dahuin Jung. LIBERO-Para: A Diagnostic Benchmark and Metrics for Paraphrase Robustness in VLA Models. *arXiv preprint arXiv:2603.28301*, 2026. URL https://arxiv.org/abs/2603.28301.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, et al. OpenVLA: An Open-Source Vision-Language-Action Model. *arXiv preprint arXiv:2406.09246*, 2024. URL https://arxiv.org/abs/2406.09246.

Moo Jin Kim, Chelsea Finn, and Percy Liang. Fine-Tuning Vision-Language-Action Models: Optimizing Speed and Success. *arXiv preprint arXiv:2502.19645*, 2025. URL https://arxiv.org/abs/2502.19645.

Jason Lee, Jiafei Duan, Haoquan Fang, Yuquan Deng, Shuo Liu, Boyang Li, Bohan Fang, Jieyu Zhang, Yi Ru Wang, Sangho Lee, Winson Han, Wilbert Pumacay, Angelica Wu, Rose Hendrix, Karen Farley, Eli VanderBilt, Ali Farhadi, Dieter Fox, and Ranjay Krishna. MolmoAct: Action Reasoning Models that can Reason in Space. *arXiv preprint arXiv:2508.07917*, 2025. URL https://arxiv.org/abs/2508.07917.

Chengshu Li, Ruohan Zhang, Josiah Wong, Cem Gokmen, Sanjana Srivastava, Roberto Martín-Martín, Chen Wang, Gabrael Levine, Wensi Ai, Benjamin Martinez, et al. BEHAVIOR-1K: A Human-Centered, Embodied AI Benchmark with 1,000 Everyday Activities and Realistic Simulation. *arXiv preprint arXiv:2403.09227*, 2024a. URL https://arxiv.org/abs/2403.09227.

Lin Li, Qihang Zhang, Yiming Luo, Shuai Yang, Ruilin Wang, Fei Han, Mingrui Yu, Zelin Gao, Nan Xue, Xing Zhu, Yujun Shen, and Yinghao Xu. Causal World Modeling for Robot Control. *arXiv preprint arXiv:2601.21998*, 2026. URL https://arxiv.org/abs/2601.21998.

Xinghang Li, Peiyan Li, Long Qian, Minghuan Liu, Dong Wang, Jirong Liu, Bingyi Kang, Xiao Ma, Xinlong Wang, Di Guo, Tao Kong, Hanbo Zhang, and Huaping Liu. What Matters in Building Vision-Language-Action Models for Generalist Robots. *arXiv preprint arXiv:2412.14058*, 2024b. URL https://arxiv.org/abs/2412.14058.

Xuanlin Li, Kyle Hsu, Jiayuan Gu, Karl Pertsch, Oier Mees, Homer Rich Walke, et al. Evaluating Real-World Robot Manipulation Policies in Simulation. *arXiv preprint arXiv:2405.05941*, 2024c. URL https://arxiv.org/abs/2405.05941.

Zhixuan Liang, Yizhuo Li, Tianshuo Yang, Chengyue Wu, Sitong Mao, Tian Nian, Liuao Pei, Shunbo Zhou, Xiaokang Yang, Jiangmiao Pang, Yao Mu, and Ping Luo. Discrete Diffusion VLA: Bringing Discrete Diffusion to Action Decoding in Vision-Language-Action Policies. *arXiv preprint arXiv:2508.20072*, 2025. URL https://arxiv.org/abs/2508.20072.

Minghui Lin, Pengxiang Ding, Shu Wang, Zifeng Zhuang, Yang Liu, Xinyang Tong, Wenxuan Song, Shangke Lyu, Siteng Huang, and Donglin Wang. HiF-VLA: Hindsight, Insight and Foresight through Motion Representation for Vision-Language-Action Models. *arXiv preprint arXiv:2512.09928*, 2025. URL https://arxiv.org/abs/2512.09928.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow Matching for Generative Modeling. *arXiv preprint arXiv:2210.02747*, 2022. URL https://arxiv.org/abs/2210.02747.

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. URL https://arxiv.org/abs/2306.03310.

Yicheng Liu, Zibin Dong, Baijun Ye, Tianyuan Yuan, Tao Jiang, Anqi Yang, Shicheng Cao, Haonan Liu, Yue Sun, Zihan Guo, Xiao Liu, Dong Ke, Changxun Pan, Chenru Wu, Tailai Cheng, Xiaoshu Ren, Xinlei Zhang, Jianning Cui, Zijie Zhao, Haoyu Zhang, Kaiming Xu, Haodong Yang, Bowen Zhang, Jiahui Niu, Shaoting Zhu, Shiduo Zhang, and Hang Zhao. G0.5: One Autoregressive Stream for Robot Reasoning and Action. *arXiv preprint arXiv:2608.11739*, 2026. URL https://arxiv.org/abs/2608.11739.

Yuankai Luo, Woping Chen, Tong Liang, Baiqiao Wang, and Zhenguo Li. SimVLA: A Simple VLA Baseline for Robotic Manipulation. *arXiv preprint arXiv:2602.18224*, 2026. URL https://arxiv.org/abs/2602.18224.

Corey Lynch and Pierre Sermanet. Language Conditioned Imitation Learning over Unstructured Data. In *Proceedings of Robotics: Science and Systems*, Virtual, July 2021. doi: 10.15607/RSS.2021.XVII.047. URL https://arxiv.org/abs/2005.07648.

Oier Mees, Lukas Hermann, and Wolfram Burgard. What Matters in Language Conditioned Robotic Imitation Learning over Unstructured Data. *IEEE Robotics and Automation Letters*, 7(4):11205–11212, 2022a. doi: 10.1109/LRA.2022.3196123. URL https://arxiv.org/abs/2204.06252.

Oier Mees, Lukas Hermann, Erick Rosete-Beas, and Wolfram Burgard. CALVIN: A Benchmark for Language-Conditioned Policy Learning for Long-Horizon Robot Manipulation Tasks. *IEEE Robotics and Automation Letters*, 7(3):7327–7334, 2022b. doi: 10.1109/LRA.2022.3180108. URL https://arxiv.org/abs/2112.03227.

NVIDIA, Johan Bjorck, Fernando Castañeda, Nikita Cherniadev, Xingye Da, Runyu Ding, Linxi "Jim" Fan, Yu Fang, Dieter Fox, Fengyuan Hu, Spencer Huang, Joel Jang, Zhenyu Jiang, Jan Kautz, Kaushil Kundalia, Lawrence Lao, Zhiqi Li, Zongyu Lin, Kevin Lin, Guilin Liu, Edith Llontop, Loic Magne, Ajay Mandlekar, Avnish Narayan, Soroush Nasiriany, Scott Reed, You Liang Tan, Guanzhi Wang, Zu Wang, Jing Wang, Qi Wang, Jiannan Xiang, Yuqi Xie, Yinzhen Xu, Zhenjia Xu, Seonghyeon Ye, Zhiding Yu, Ao Zhang, Hao Zhang, Yizhou Zhao, Ruijie Zheng, and Yuke Zhu. GR00T N1: An Open Foundation Model for Generalist Humanoid Robots. *arXiv preprint arXiv:2503.14734*, 2025. URL https://arxiv.org/abs/2503.14734.

NVIDIA Research. Comet: 2025 BEHAVIOR Challenge Public Validation entry (rank 4, Standard track). 2025 BEHAVIOR Challenge public leaderboard, 2025.

Octo Model Team, Dibya Ghosh, Homer Walke, Karl Pertsch, Kevin Black, Oier Mees, Sudeep Dasari, Joey Hejna, Tobias Kreiman, Charles Xu, et al. Octo: An Open-Source Generalist Robot Policy. *arXiv preprint arXiv:2405.12213*, 2024. URL https://arxiv.org/abs/2405.12213.

Open X-Embodiment Collaboration, Abby O'Neill, Abdul Rehman, Abhinav Gupta, Abhiram Maddukuri, Abhishek Gupta, Abhishek Padalkar, Abraham Lee, Acorn Pooley, Agrim Gupta, et al. Open X-Embodiment: Robotic Learning Datasets and RT-X Models. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 6892–6903. IEEE, 2024. doi: 10.1109/ICRA57147.2024.10611477. URL https://arxiv.org/abs/2310.08864.

Karl Pertsch, Kyle Stachowicz, Brian Ichter, Danny Driess, Suraj Nair, Quan Vuong, et al. FAST: Efficient Action Tokenization for Vision-Language-Action Models. *arXiv preprint arXiv:2501.09747*, 2025. URL https://arxiv.org/abs/2501.09747.

Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, Danny Driess, Chelsea Finn, Karol Hausman, et al. $\pi_{0.5}$: A Vision-Language-Action Model with Open-World Generalization. *arXiv preprint arXiv:2504.16054*, 2025. URL https://arxiv.org/abs/2504.16054.

Delin Qu, Haoming Song, Qizhi Chen, Yuanqi Yao, Xinyi Ye, Yan Ding, Zhigang Wang, Jiayuan Gu, Bin Zhao, Dong Wang, and Xuelong Li. SpatialVLA: Exploring Spatial Representations for Visual-Language-Action Model. *arXiv preprint arXiv:2501.15830*, 2025. URL https://arxiv.org/abs/2501.15830.

Moritz Reuss, Hongyi Zhou, Marcel Rühle, Ömer Erdinç Yağmurlu, Fabian Otto, and Rudolf Lioutikov. FLOWER: Democratizing Generalist Robot Policies with Efficient Vision-Language-Action Flow Policies. *arXiv preprint arXiv:2509.04996*, 2025. URL https://arxiv.org/abs/2509.04996.

Robot Learning Collective. Robot Learning Collective: 2025 BEHAVIOR Challenge Public Validation entry (rank 1, Standard track). 2025 BEHAVIOR Challenge public leaderboard. Affiliation: Independent, 2025.

Qiuhong Shen, Shihua Zhang, Yue Liao, Qi Li, Zhenxiong Tan, Shizun Wang, Shuicheng Yan, and Xinchao Wang. World Action Models: A Survey. *arXiv preprint arXiv:2606.20781*, 2026. URL https://arxiv.org/abs/2606.20781.

Hao Shi, Bin Xie, Yingfei Liu, Lin Sun, Fengrong Liu, Tiancai Wang, Erjin Zhou, Haoqiang Fan, Xiangyu Zhang, and Gao Huang. MemoryVLA: Perceptual-Cognitive Memory in Vision-Language-Action Models for Robotic Manipulation. *arXiv preprint arXiv:2508.19236*, 2025. URL https://arxiv.org/abs/2508.19236.

Mustafa Shukor, Dana Aubakirova, Francesco Capuano, Pepijn Kooijmans, Steven Palma, Adil Zouitine, et al. SmolVLA: A Vision-Language-Action Model for Affordable and Efficient Robotics. *arXiv preprint arXiv:2506.01844*, 2025. URL https://arxiv.org/abs/2506.01844.

Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, Francisco Massa, Daniel Haziza, Luca Wehrstedt, Jianyuan Wang, Timothée Darcet, Théo Moutakanni, Leonel Sentana, Claire Roberts, Andrea Vedaldi, Jamie Tolan, John Brandt, Camille Couprie, Julien Mairal, Hervé Jégou, Patrick Labatut, and Piotr Bojanowski. DINOv3. *arXiv preprint arXiv:2508.10104*, 2025. URL https://arxiv.org/abs/2508.10104.

Wenxuan Song, Jiayi Chen, Pengxiang Ding, Han Zhao, Wei Zhao, Zhide Zhong, Zongyuan Ge, Zhijun Li, Donglin Wang, Jun Ma, Lujia Wang, and Haoang Li. PD-VLA: Accelerating Vision-Language-Action Model Integrated with Action Chunking via Parallel Decoding. *arXiv preprint arXiv:2503.02310*, 2025. URL https://arxiv.org/abs/2503.02310.

StarVLA Community. StarVLA: A Lego-like Codebase for Vision-Language-Action Model Developing. *arXiv preprint arXiv:2604.05014*, 2026. URL https://arxiv.org/abs/2604.05014.

Yang Tian, Sizhe Yang, Jia Zeng, Ping Wang, Dahua Lin, Hao Dong, and Jiangmiao Pang. Predictive Inverse Dynamics Models are Scalable Learners for Robotic Manipulation. *arXiv preprint arXiv:2412.15109*, 2024. URL https://arxiv.org/abs/2412.15109.

Michael Tschannen, Alexey Gritsenko, Xiao Wang, Muhammad Ferjad Naeem, Ibrahim Alabdulmohsin, Nikhil Parthasarathy, Talfan Evans, Lucas Beyer, Ye Xia, Basil Mustafa, Olivier Hénaff, Jeremiah Harmsen, Andreas Steiner, and Xiaohua Zhai. SigLIP 2: Multilingual Vision-Language Encoders with Improved Semantic Understanding, Localization, and Dense Features. *arXiv preprint arXiv:2502.14786*, 2025. URL https://arxiv.org/abs/2502.14786.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, et al. Attention Is All You Need. In *Advances in Neural Information Processing Systems*, 2017. URL https://arxiv.org/abs/1706.03762.

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, Abraham Lee, Kuan Fang, Chelsea Finn, and Sergey Levine. BridgeData V2: A Dataset for Robot Learning at Scale. In *Proceedings of the 7th Conference on Robot Learning*, volume 229 of *Proceedings of Machine Learning Research*, pp. 1723–1736. PMLR, 2023. URL https://proceedings.mlr.press/v229/walke23a.html.

Yihao Wang, Pengxiang Ding, Lingxiao Li, Can Cui, Zirui Ge, Xinyang Tong, et al. VLA-Adapter: An Effective Paradigm for Tiny-Scale Vision-Language-Action Model. *arXiv preprint arXiv:2509.09372*, 2025a. URL https://arxiv.org/abs/2509.09372.

Yuqi Wang, Xinghang Li, Wenxuan Wang, Junbo Zhang, Yingyan Li, Yuntao Chen, Xinlong Wang, and Zhaoxiang Zhang. Unified Vision-Language-Action Model. *arXiv preprint arXiv:2506.19850*, 2025b. URL https://arxiv.org/abs/2506.19850.

Wei Wu, Fan Lu, Yunnan Wang, Shuai Yang, Shi Liu, Fangjing Wang, Qian Zhu, He Sun, Yong Wang, Shuailei Ma, Yiyu Ren, Kejia Zhang, Hui Yu, Jingmei Zhao, Shuai Zhou, Zhenqi Qiu, Houlong Xiong, Ziyu Wang, Zechen Wang, Ran Cheng, Yong-Lu Li, Yongtao Huang, Xing Zhu, Yujun Shen, and Kecheng Zheng. A Pragmatic VLA Foundation Model. *arXiv preprint arXiv:2601.18692*, 2026a. URL https://arxiv.org/abs/2601.18692.

Xiao-Ming Wu, Bin Fan, Kang Liao, Jian-Jian Jiang, Runze Yang, Yihang Luo, Zhonghua Wu, Wei-Shi Zheng, and Chen Change Loy. VLANeXt: Recipes for Building Strong VLA Models. *arXiv preprint arXiv:2602.18532*, 2026b. URL https://arxiv.org/abs/2602.18532.

Hengyi Xie, Chenfei Yao, Xianjin Wu, Yingying Zhu, Dingkang Liang, Xiang Bai, and Han Ding. TurboVLA: Real-Time Vision-Language-Action Model at 32 Hz on an RTX 4090 with $<$1 GB VRAM. *arXiv preprint arXiv:2607.27205*, 2026. URL https://arxiv.org/abs/2607.27205.

Yifan Yang, Zhixiang Duan, Tianshi Xie, Fuyu Cao, Pinxi Shen, Peili Song, Piaopiao Jin, Guokang Sun, Shaoqing Xu, Yangwei You, and Jingtai Liu. FPC-VLA: A Vision-Language-Action Framework with a Supervisor for Failure Prediction and Correction. *arXiv preprint arXiv:2509.04018*, 2025. URL https://arxiv.org/abs/2509.04018.

Zhaokai Yin and Zhipeng Zhang. Grounded Semantic Re-Binding for Robust Instruction Generalization in Vision-Language-Action Models. *arXiv preprint arXiv:2608.02497*, 2026. URL https://arxiv.org/abs/2608.02497.

Tianyuan Yuan, Zibin Dong, Yicheng Liu, and Hang Zhao. Fast-WAM: Do World Action Models Need Test-time Future Imagination? *arXiv preprint arXiv:2603.16666*, 2026. URL https://arxiv.org/abs/2603.16666.

Jiahui Zhang, Yurui Chen, Yueming Xu, Ze Huang, Yanpeng Zhou, Yu-Jie Yuan, Xinyue Cai, Guowei Huang, Xingyue Quan, Hang Xu, and Li Zhang. 4D-VLA: Spatiotemporal Vision-Language-Action Pretraining with Cross-Scene Calibration. *arXiv preprint arXiv:2506.22242*, 2025. URL https://arxiv.org/abs/2506.22242.

Jianke Zhang, Yuanfei Luo, Yucheng Hu, Xiaoyu Chen, Yanjiang Guo, Ziyang Liu, Hongbin Xu, Tian Lan, and Jianyu Chen. UAM: A Dual-Stream Perspective on Forgetting in VLA Training. *arXiv preprint arXiv:2605.15735*, 2026a. URL https://arxiv.org/abs/2605.15735.

Qihang Zhang, Lin Li, Luyao Zhang, Shuai Yang, Yiming Luo, Shuaiting Li, Ruilin Wang, Junke Wang, Jiahao Shao, Gangwei Xu, Jiaming Zhou, Yishu Shen, Yudong Jin, Fangyi Xu, Shuailei Ma, Jiaqi Liao, Guanxing Lu, Zifan Shi, Yongkun Wen, Yujie Zhao, Weixuan Tang, Xinyang

Wang, Chaojian Li, Jiapeng Zhu, Ka Leong Cheng, Nan Xue, Xing Zhu, Yujun Shen, and Yinghao Xu. Native Video-Action Pretraining for Generalizable Robot Control. *arXiv preprint arXiv:2607.08639*, 2026b. URL https://arxiv.org/abs/2607.08639.

Qingqing Zhao, Yao Lu, Moo Jin Kim, Zipeng Fu, Zhuoyang Zhang, Yecheng Wu, Zhaoshuo Li, Qianli Ma, Song Han, Chelsea Finn, Ankur Handa, Ming-Yu Liu, Donglai Xiang, Gordon Wetzstein, and Tsung-Yi Lin. CoT-VLA: Visual Chain-of-Thought Reasoning for Vision-Language-Action Models. *arXiv preprint arXiv:2503.22020*, 2025. URL https://arxiv.org/abs/2503.22020.

Jinliang Zheng, Jianxiong Li, Dongxiu Liu, Yinan Zheng, Zhihao Wang, Zhonghong Ou, Yu Liu, Jingjing Liu, Ya-Qin Zhang, and Xianyuan Zhan. Universal Actions for Enhanced Embodied Foundation Models. *arXiv preprint arXiv:2501.10105*, 2025a. URL https://arxiv.org/abs/2501.10105.

Jinliang Zheng, Jianxiong Li, Zhihao Wang, Dongxiu Liu, Xirui Kang, Yuchun Feng, et al. X-VLA: Soft-Prompted Transformer as Scalable Cross-Embodiment Vision-Language-Action Model. *arXiv preprint arXiv:2510.10274*, 2025b. URL https://arxiv.org/abs/2510.10274.

Ruijie Zheng, Yongyuan Liang, Shuaiyi Huang, Jianfeng Gao, Hal Daumé III, Andrey Kolobov, Furong Huang, and Jianwei Yang. TraceVLA: Visual Trace Prompting Enhances Spatial-Temporal Awareness for Generalist Robotic Policies. *arXiv preprint arXiv:2412.10345*, 2024. URL https://arxiv.org/abs/2412.10345.

Zhide Zhong, Haodong Yan, Junfeng Li, Xiangchen Liu, Xin Gong, Tianran Zhang, Wenxuan Song, Jiayi Chen, Xinhu Zheng, Hesheng Wang, and Haoang Li. FlowVLA: Visual Chain of Thought-based Motion Reasoning for Vision-Language-Action Models. *arXiv preprint arXiv:2508.18269*, 2025. URL https://arxiv.org/abs/2508.18269.

Xueyang Zhou, Yangming Xu, Guiyao Tie, Yongchao Chen, Guowen Zhang, Duanfeng Chu, Pan Zhou, and Lichao Sun. LIBERO-PRO: Towards Robust and Fair Evaluation of Vision-Language-Action Models Beyond Memorization. *arXiv preprint arXiv:2510.03827*, 2025. URL https://arxiv.org/abs/2510.03827.

A Implementation Details

A.1 Evaluation Protocols

Table 10 summarizes the evaluator setting and metric used for each benchmark evaluation.

Table 10: **Benchmark-specific evaluation protocols.**

Evaluation	Protocol	Metric
LIBERO	Four ten-task suites; predict $H{=}10$, execute $h{=}5$	Suite and macro success (%)
LIBERO-PRO	Original plus object, position, semantic, and task perturbations	Per-suite and perturbation success (%)
CALVIN ABC→D	1,000 five-subtask chains in held-out scene D	SR@1–5, mean chain length
WidowX	SimplerEnv-Bridge, four manipulation tasks	Per-task success (%), average
Google Robot	SimplerEnv-Fractal variant aggregation	Variant-aggregation success (%)
RoboTwin 2.0	Bimanual Easy/Hard evaluation	Success (%)
BEHAVIOR-1K	50 household behaviors, 10 rollouts each	Q-Score

A.2 Domain Interfaces and Action Spaces

Table 11 makes explicit what is and is not shared across domains. All runs use the high-level {images, view mask, instruction codes, proprioception, action} interface, with benchmark-specific dataset statistics, control adapters, and evaluators.

Table 11: **Benchmark-specific control interfaces.** H is predicted chunk length, h is maximum execution before replanning, r is execution stride, and K is the Euler-step count.

Domain	Views	State	Native/model action	$H/h/r/K$	Control semantics
LIBERO/LIBERO-PRO	2	8D	7D/7D	10/5/1/10	Delta EE pose/gripper
CALVIN	2	20D	10D/20D	30/30/1/20	10D active arm + 10D zero dummy; execute first 10D
WidowX	1	7D	7D/7D	30/30/1/30	Absolute EE position/Euler target
Google Robot	1	zero 7D	7D/7D	30/10/2/30	Accumulated translation; absolute orientation
RoboTwin 2.0	3	14D	14D/14D	125/50/1/10	Absolute dual-arm joint targets
BEHAVIOR-1K	3	23D	23D/23D	30/30/1/10	Relative targets for trunk/arms; absolute base velocity and grippers

A.3 Training Hyperparameters

Table 12 provides the benchmark-specific schedules used for the reported manipulation results. Training is iteration-based: because trajectories have different frame-level lengths across datasets, a nominal epoch is not directly comparable across domains, so we report the total number of training iterations.

Table 12: **Benchmark-specific training hyperparameters.** §BEHAVIOR-1K is trained in two stages: a 50-task multi-task pretraining run, followed by a fine-tuning run at the lower learning rate.

Configuration	LIBERO/LIBERO-PRO	CALVIN	WidowX	Google	RoboTwin	BEHAVIOR-1K
Image size / valid views	256 / 2	256 / 2	256 / 1	256 / 1	256 / 3	256 / 3
Action head d_{model}/layers/heads	1024/24/16	1024/24/16	1024/24/16	1024/24/16	1024/24/16	1024/24/16
Global batch	256	128	256	256	160	512
Base learning rate	$5{\times}10^{-4}$	10^{-4}	10^{-4}	10^{-4}	$2{\times}10^{-4}$	10^{-4} / 10^{-5} §
Vision LR multiplier	0.1	0.1	0.05	0.05	0.1	0.01
Training iterations	200k	200k	100k	200k	200k	200k§
Eval checkpoint	200k	200k	100k	200k	200k	200k§
Precision	bfloat16 activation autocast; FP32 trainable parameter storage					
Language model	None: fixed codes + random action-trained lookup					

A.4 Flow-Matching Instantiation of the Action Head

Every reported run instantiates the head of Section 3 with conditional flow matching (Lipman et al., 2022). For a normalized expert chunk x and $\epsilon \sim \mathcal{N}(0, I)$, we sample $\widetilde{\tau} \sim \mathrm{Beta}(1.5, 1)$, set $\tau = 0.999\widetilde{\tau} + 0.001$, and construct

$$x_\tau = \tau\epsilon + (1-\tau)x, \qquad u_\tau = \epsilon - x,$$



Figure 2: **BEHAVIOR-1K rollouts.** Eight household activities, one goal-satisfying episode each (q=1), with six frames sampled across the episode from left to right. Each frame shows the three onboard views the policy receives: egocentric, left wrist, and right wrist. Episode lengths differ by an order of magnitude across these rows, from roughly 2,100 simulation steps for *turning on radio* to 14,700 for *setting the fire*.

so that the auxiliary input is $z = (\tau, x_\tau)$ and the target is $y = u_\tau$. The training objective is

$$\mathcal{L}_{\mathrm{FM}} = \mathbb{E}_{x,\epsilon,\tau}\left[\|v_\theta(x_\tau, c_t, \tau) - u_\tau\|_2^2\right].$$

Inference initializes $x_1 \sim \mathcal{N}(0, I)$ and applies explicit Euler integration from $\tau = 1$ to 0 using the K steps recorded in Table 11.

A.5 Parameter Accounting

Table 13: **Approximate architectural parameter accounting for the LM-free SigLIP2 models.** The table separates the vision backbone, action-trained instruction embedding, action Transformer, and pretrained language-side parameters.

Component	Manipulation default
SigLIP2 vision model	$\sim$ 303M
Random instruction embedding	$\sim$ 31M
Action Transformer	$\sim$ 303M
Total incl. projections/heads	$\sim$ 639M
Pretrained language-side parameters	**0**

SIMVLA 2.0 is not a small policy in total parameter count. Its approximately 31M instruction-embedding parameters are random at initialization and action-trained, not pretrained language parameters. With FP32 parameters, gradients, and two Adam moments, the 31M instruction-embedding table alone occupies approximately 0.5 decimal GB of training state before framework and sharding overhead.

A.6 BEHAVIOR-1K QUALITATIVE ROLLOUTS

Figure 2 shows the executed behavior on eight such activities.

A.7 SEED VARIANCE: A NOISE FLOOR FOR THE ABLATIONS

We train the default configuration with three seeds (0, 1, 2) on LIBERO and on WidowX, evaluate each under the same strict protocol as the main results (500 episodes per LIBERO suite, 24 episodes per WidowX task), and take the resulting range and standard deviation as the noise floor.

Table 14: **LIBERO default-recipe seed variance.** Success (%), 500 episodes per suite.

Seed	Spatial	Object	Goal	Long	Avg
0	98.6	99.0	96.4	91.4	96.4
1	98.4	99.6	97.4	92.2	96.9
2	98.2	98.6	96.0	91.8	96.2
Mean $\pm$ std	98.4 ± 0.2	99.1 ± 0.4	96.6 ± 0.6	91.8 ± 0.3	$\mathbf{96.5 \pm 0.3}$

Table 15: **WidowX default-recipe seed variance.** Success (%), 24 episodes per task.

Seed	Stack	Eggplant	Carrot	Spoon	Avg
0	79.2	87.5	100.0	100.0	91.7
1	79.2	83.3	100.0	100.0	90.6
2	91.7	87.5	100.0	100.0	94.8
Mean $\pm$ std	83.3 ± 5.9	86.1 ± 2.0	100.0 ± 0.0	100.0 ± 0.0	$\mathbf{92.4 \pm 1.8}$

LIBERO is tight (range 0.8 points, std 0.3); WidowX is far noisier (range 4.2 points, std 1.8), driven almost entirely by the Stack task.

A.8 FURTHER DISCUSSION

Reproducibility Statement Section 4.1 summarizes the benchmark suite and evaluation setup. The appendix provides evaluation protocols, benchmark-specific hyperparameters, control interfaces, and flow-matching implementation details (Table 10; Table 12; Table 11; Section A.4). Code is available at https://anonymous.4open.science/r/SimVLA2-4E2F.

Ethics Statement The reported benchmarks are simulated or benchmark-mediated robot-control evaluations and contain no human-subject data. Benchmark results do not establish safe deployment; physical robot systems require collision handling, uncertainty monitoring, workspace limits, hardware interlocks, and human oversight.