IntentVLA: Short-Horizon Intent Modeling for Aliased Robot Manipulation

Anonymous ACL submission

Abstract

Robot imitation data are often multimodal: similar visual-language observations may be followed by different action chunks because human demonstrators act with different short-horizon intents, task phases, or recent context. Existing frame-conditioned VLA policies infer each chunk from the current observation and instruction alone, so under partial observability they may resample different intents across adjacent replanning steps, leading to inter-chunk conflict and unstable execution. We introduce IntentVLA, a history-conditioned VLA framework that encodes recent visual observations into a compact short-horizon intent representation and uses it to condition chunk generation. We further introduce AliasBench, a 12-task ambiguity-aware benchmark on RoboTwin2 with matched training data and evaluation environments that isolate short-horizon observation aliasing. Across AliasBench, SimplerEnv, LIBERO, and RoboCasa, IntentVLA improves rollout stability and outperforms strong VLA baselines. The benchmark code, generated data, and model code are released at `https://anonymous.4open.science/r/IntentVLA`.

1 Introduction

Vision-language-action (VLA) models provide a direct interface from perception and instruction to control: given visual observations and a language command, the policy outputs robot actions (Kim et al., 2024; Black et al., 2024; Liu et al., 2025; Bjorck et al., 2025). Recent large-model-based VLAs scale this paradigm with transformer backbones, large robot datasets, and vision-language pretraining, enabling more generalist manipulation policies across tasks and embodiments (Intelligence et al., 2025; GEAR-Team et al., 2025; Bi et al., 2025; Zheng et al., 2025a; Liu et al., 2026).

Training VLA models typically relies on large-scale human-collected robot trajectories (O'Neill et al., 2024; Bu et al., 2025a; Walke et al., 2023; Khazatsky et al., 2024), and these datasets often faithfully reflect the underlying multimodality of manipulation behavior. For instance, an environment may admit multiple valid goals, and even a fixed goal can often be achieved through multiple feasible paths (Zhai et al., 2025a). This diversity is not itself the problem. Human demonstrations are naturally multimodal across episodes, but they are locally committed within each episode: once a demonstrator follows a particular task phase, path, or completion strategy, adjacent action chunks usually remain consistent with that choice. The difficulty arises because current VLA policies generally infer actions from only the current frame image and the language instruction. Under partial observability, the same frame-level observation can correspond to different short-horizon intents, but a frame-conditioned VLA does not observe the episode-level commitment that selected one of them. Repeated chunk generation can then switch among intents across adjacent decision steps, producing contradictory chunks and unstable execution. Thus, the goal is not to eliminate multimodality, but to condition generation on the commitment already expressed by the current episode.

Figure 1 illustrates this ambiguity: similar bread-holding observations can require skillet placement or plate return under the same instruction. To measure this failure mode, we build AliasBench on RoboTwin2 (Chen et al., 2025) with matched simulation training data and evaluation environments that isolate short-horizon observation aliasing. Rather than only scoring task success, AliasBench tests whether policies preserve local continuations when visually similar states require different next chunks. It covers back-and-forth, crossing-path, bimanual, and multi-goal ambiguity, with examples in Figure 2.

To address the failure mode validated by AliasBench, we propose IntentVLA, a history-conditioned imitation learning framework for chun-

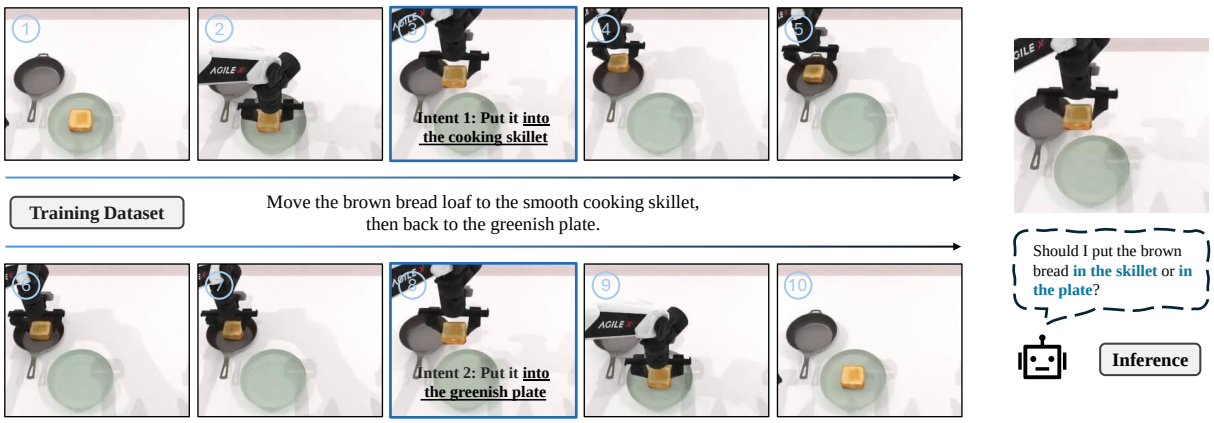

Figure 1: **Short-horizon intent ambiguity under frame-only conditioning.** Similar bread-in-gripper observations require different continuations: cooking in the skillet or returning to the plate.

ked VLA control. The core idea is to preserve the episode's local commitment by conditioning action generation on recent visual evidence, rather than inferring every chunk from the current frame alone. Concretely, IntentVLA encodes recent observations with a frozen VGGT-based history encoder, keeps compact camera and register tokens as history evidence, and fuses them with the current Qwen3-VL visual-language context through gated cross-attention. The fused context and an appended history-evidence token form a short-horizon intent representation that conditions a standard DiT-based flow-matching action head. In experiments on AliasBench, SimplerEnv, LIBERO, and RoboCasa, IntentVLA improves both success rate and execution stability over strong VLA baselines.

Our contributions are fourfold:

- We identify a failure mode of frame-conditioned chunk policies under partial observability: demonstrations are multimodal across episodes but locally committed within an episode, while frame-only conditioning can break this commitment at test time.
- We construct AliasBench, a 12-task benchmark on RoboTwin2 for evaluating VLA behavior under short-horizon observation aliasing, together with matched simulation training data and evaluation environments.
- We propose IntentVLA, a history-conditioned imitation learning framework that learns a compact short-horizon intent representation from recent visual observations and uses it to condition chunk generation.
- We implement and validate IntentVLA extensively across AliasBench, SimplerEnv, LIBERO, and RoboCasa, including ambiguous-intent tasks that directly test short-horizon intent consistency.

2 Related Work

2.1 Vision-Language-Action Models

Vision-Language-Action (VLA) models connect vision-language pre-training with robot control. RT-2 (Zitkovich et al., 2023) and OpenVLA (Kim et al., 2024) transfer VLM semantic priors to action generation, while FAST (Pertsch et al., 2025) improves efficiency through frequency-space compression. For continuous multi-step control, Octo (Team et al., 2024), π_0 (Black et al., 2024), and RDT-1B (Liu et al., 2025) use diffusion or flow-matching action heads, and $\pi_{0.5}$ (Intelligence et al., 2025) scales this direction with heterogeneous data and multimodal supervision. Other work addresses robot-data scarcity with synthetic data or human videos (Bjorck et al., 2025; Bu et al., 2025b), improves cross-embodiment transfer through prompt-based adaptation (Bi et al., 2025; Zheng et al., 2025a), strengthens spatial grounding with 3D-aware representations (Qu et al., 2025; Zhen et al., 2024; Li et al., 2025b; Lin et al., 2025a; Li et al., 2025a, 2026a), or adds world-model, future-prediction, and memory mechanisms for longer-horizon reasoning (Shen et al., 2025; Sun et al., 2026; Shi et al., 2026; Chen et al., 2026a).

2.2 Intent-based VLA Models

Intent-driven VLAs aim to bridge semantic reasoning and low-level control. DIAL (Chen et al., 2026b) uses a differentiable latent intent bottleneck with visual foresight, and ACoT-VLA (Zhong

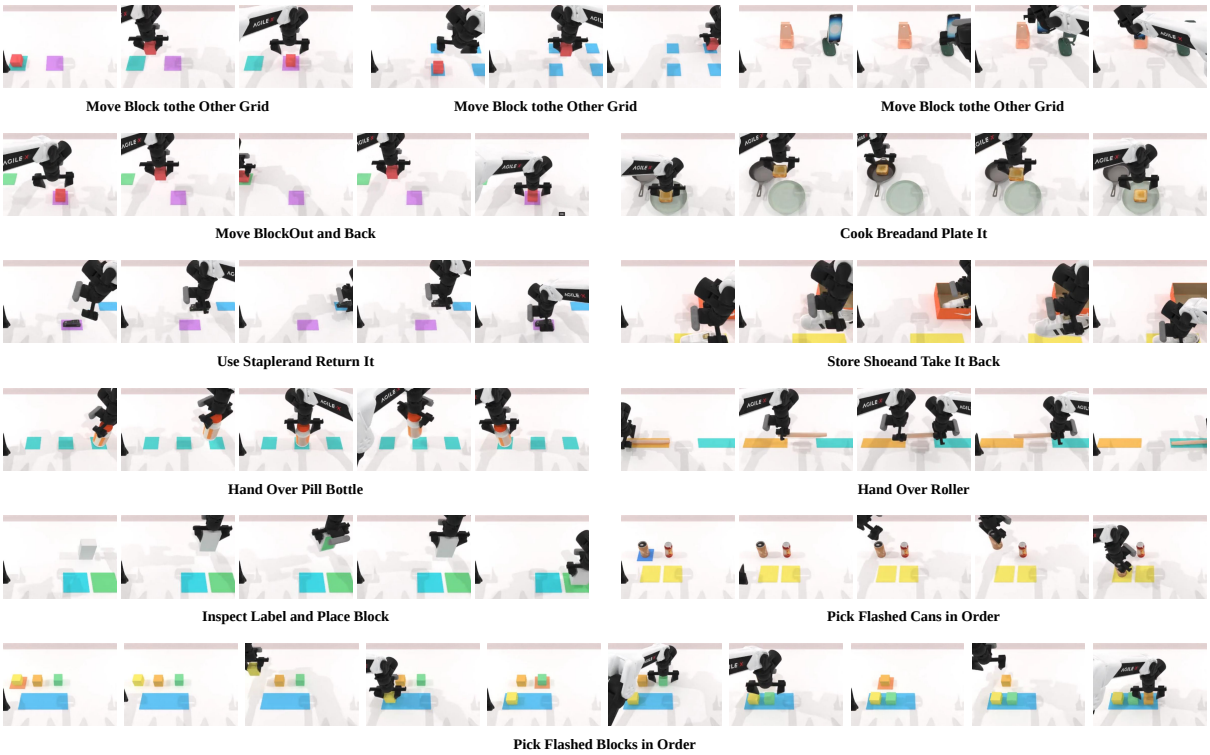

Figure 2: **Representative observation aliasing patterns in AliasBench.** The quantitative observation-aliasing diagnostic is shown in Figure 3, and policy results are reported in Section 5.1.

et al., 2026) represents reasoning as kinematically grounded action intents. GazeVLA (Li et al., 2026b) models human intention through gaze as an intermediate representation. MINT (Huang et al., 2026) disentangles low-frequency intent from high-frequency execution residuals, MAIN-VLA (Zhou et al., 2026) compresses instructions into semantic primitives with structured affordances, DeepVision-VLA (Luo et al., 2026) uses action-guided pruning for task-relevant visual grounding, and VFP (Zhai et al., 2025b) models multimodal expert behavior with a variational latent prior. However, these methods rely on the current observation frame, often struggling to resolve short-horizon ambiguity under partial observability where visually similar states require different continuations that can only be disambiguated by recent task history.

3 AliasBench: Ambiguity-Aware Benchmark Design

To evaluate whether a policy can resolve aliased observations from recent context, we build AliasBench on top of RoboTwin2 (Chen et al., 2025). AliasBench contains 12 manipulation tasks together with matched simulation training data and held-out evaluation environments. The benchmark targets an underexplored gap in current VLA evaluation: most standard benchmarks measure whether a policy can complete a manipulation task, but they rarely isolate whether the policy can maintain a consistent decision when the current observation is aliased. AliasBench is therefore designed as a tool for testing whether VLAs can preserve decision consistency across adjacent action chunks. Concretely, we seek task configurations in which two episode states produce nearly identical current observations,

$$o_t^{(1)} \approx o_t^{(2)}, \tag{1}$$

but require different next actions,

$$a_t^{(1)} \neq a_t^{(2)}. \tag{2}$$

The difference should arise from latent context that is not identifiable from the current frame alone but is still recoverable from recent observations. This is the failure mode we want the benchmark to expose.

We group the 12 tasks by the latent factor that resolves the alias: back-and-forth phase, crossing-path source, bimanual handoff direction, and multi-goal cues. These families cover common manipulation patterns in which the current frame is visually plausible for multiple continuations. Figure 2 shows representative cases, and Appendix E provides task-level definitions and examples.

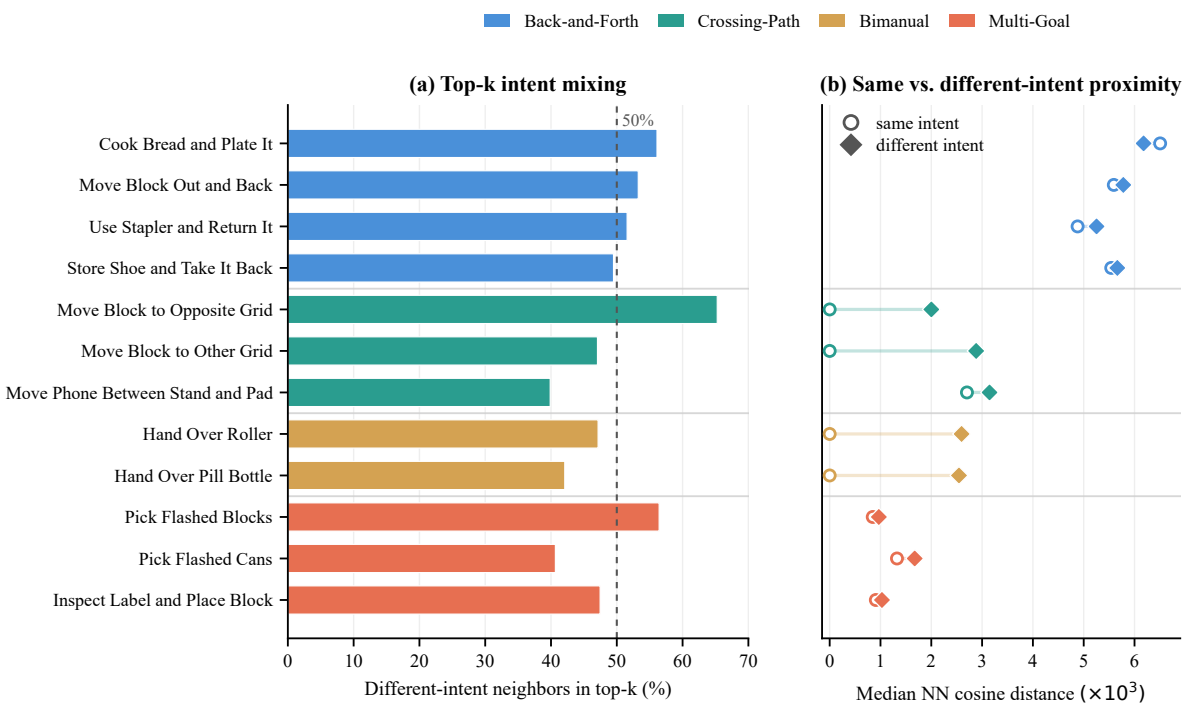

Figure 3: **Quantitative observation-aliasing diagnostic on AliasBench.** Embeddings are extracted with Qwen3.5VL-8B. Back-and-Forth uses intra-episode retrieval with a 20-frame temporal gap; all other families use cross-episode retrieval. The diagnostic is not a policy success metric. Instead, it measures whether visually nearby states in the ambiguity window can correspond to different next intents. Left: roughly half of the top-k neighbors ($k = 5$) come from a different intent. Right: open circles and filled diamonds show median nearest-neighbor cosine distances to same-intent and different-intent states, respectively, with distances scaled by 10^3 for readability. Although a few tasks appear to have larger distance gaps visually, the actual cosine-distance differences remain on the order of 10^{-3}.

Observation-aliasing diagnostic. We quantify whether these aliases are measurable using nearest-neighbor retrieval over visual embeddings inside ambiguity windows; Appendix E gives the retrieval protocol. The diagnostic is not a policy success metric, but a check that visually nearby states can correspond to different next intents.

Figure 3 confirms strong current-frame aliasing: the average different-intent neighbor ratio is 49.7%, and the largest median same-vs.-different intent gap is below 3×10^{-3} in cosine distance. Thus, AliasBench isolates states where the current frame gives weak evidence about the short-horizon intent, while recent history contains the missing context.

4 Method

4.1 Motivation and Problem Formulation

We model manipulation as a partially observable decision process with latent state $s_t \in \mathcal{S}$, observation $o_t \in \mathcal{O}$, and action $a_t \in \mathcal{A}$. This viewpoint is used only to motivate why recent observations can disambiguate the current frame. At time step t, the robot observes o_t, receives a language instruction ℓ, and predicts a future action chunk

$$\tau_t = (a_t, a_{t+1}, \ldots, a_{t+H-1}) \in \mathbb{R}^{H \times d_a}, \tag{3}$$

where H is the chunk horizon and d_a is the action dimension. Instead of using the complete interaction history $H_t = (o_{1:t}, a_{1:t-1})$, IntentVLA uses a finite visual history window $h_t^K = o_{t-K:t-1}$ as compact evidence about the recent episode context. To formalize the ambiguity, let z_t denote a latent short-horizon intent, such as a local continuation mode, task phase, or committed path. A standard frame-conditioned chunk policy models $p_\theta(\tau_t \mid o_t, \ell)$, whose imitation target can be written conceptually as

$$p_\theta(\tau_t \mid o_t, \ell) = \int p_\theta(\tau_t \mid o_t, \ell, z_t) \, p(z_t \mid o_t, \ell) \, dz_t. \tag{4}$$

The issue is not multimodality itself, but *uncommitted multimodality under aliased conditioning*: the

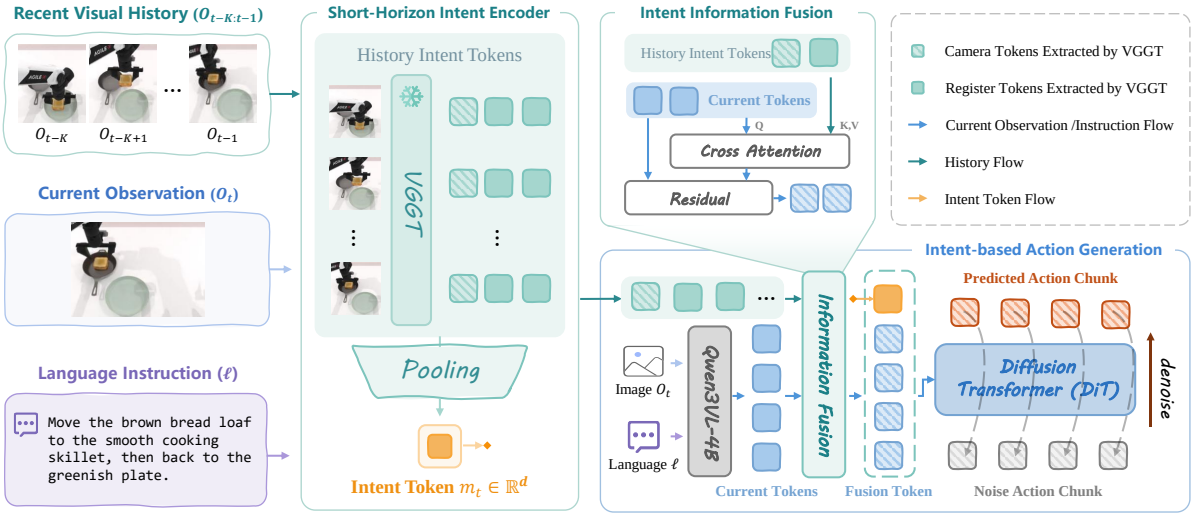

Figure 4: **Overview of IntentVLA.** A Qwen3-VL backbone encodes the current image and language instruction, while a frozen VGGT-1B history encoder extracts recent visual evidence. IntentVLA fuses the history tokens with the current visual-language context through gated cross-attention, appends a compact short-horizon intent token, and conditions a DiT-based flow-matching action head for chunk generation.

current frame and instruction may not reveal which continuation has already been selected within the episode. This motivates conditioning chunk generation on recent visual history,

$$p_\theta(\tau_t \mid o_t, \ell, h_t) = \int p_\theta(\tau_t \mid o_t, \ell, h_t, z_t) \, p(z_t \mid o_t, \ell, h_t) \, dz_t, \tag{5}$$

where h_t denotes the recent history available at time t. Rather than explicitly inferring z_t or supervising intent labels, IntentVLA learns a deterministic short-horizon intent representation $m_t = f_\phi(o_t, \ell, h_t^K)$, which serves as a compact embedding of history-conditioned intent evidence for chunk generation. Throughout the main formulation, h_t^K refers only to recent visual history. Based on this formulation, we instantiate IntentVLA as shown in Figure 4: a frozen visual-history encoder extracts recent intent evidence, a gated fusion module combines this evidence with the current VLA context, and a standard DiT-based flow-matching head generates action chunks. We describe these components below.

4.2 Short-Horizon Intent from Recent History

IntentVLA separates the current visual-language context from recent visual history. In our implementation, the current image and language instruction are processed by a Qwen3-VL 4B backbone q_ψ, whose last hidden layer gives the current-condition representation $F_t = q_\psi(o_t, \ell) \in \mathbb{R}^{N \times d}$, where N is the number of current-context tokens and d is the hidden dimension. F_t serves as the conditioning source for the action model.

In parallel, a visual history encoder processes the finite observation history window and produces both history evidence tokens and a summary representation:

$$\begin{aligned} U_t &= g_\phi(h_t^K) \in \mathbb{R}^{M \times d_h}, \\ \bar{e}_t &= \mathrm{Pool}(U_t) \in \mathbb{R}^{d_h}. \end{aligned} \tag{6}$$

Here g_ϕ is the history encoder and operates on image observations. In our method, we instantiate g_ϕ with a frozen VGGT-1B encoder (Wang et al., 2025). When each robot observation contains multiple camera views, the recent-history branch uses only the head-camera frames; the current visual-language backbone can still receive the standard current observation used by the base VLA. Specifically, we do not use all VGGT output tokens. Instead, for each input frame, we retain only this one camera token and these four register tokens. The camera token is used by VGGT for camera-parameter prediction, while the register tokens capture global geometric information and inter-frame relations. We use these tokens because they represent recent viewpoint changes and frame-to-frame structure that are particularly useful for inferring the currently active short-horizon intent. The resulting history features are then projected into the action-model hidden space:

$$\tilde{U}_t = \mathrm{LN}(W_h U_t), \quad e_t = W_e \bar{e}_t, \tag{7}$$

where W_h and W_e are learned projections and $e_t \in \mathbb{R}^d$ is a compact history-evidence token.

Accordingly, the method uses two complementary forms of history information: a sequence of fine-grained history tokens $\tilde{U}_t$ for token-level fusion, and a single compact token e_t that summarizes recent visual evidence. The compact token is not meant to be a standalone latent intent variable. It provides history evidence, while the condition-dependent intent representation is formed only after this evidence is combined with the current image-language context. All components are learned jointly with the policy objective and require no explicit supervision on intent labels.

4.3 Intent-based Action Generation

We fuse the current visual-language context F_t with the history tokens using gated cross-attention. Specifically,

$$F'_t = F_t + \sigma(\alpha)\,\mathrm{MHA}(Q = \mathrm{LN}(F_t), K = \tilde{U}_t, V = \tilde{U}_t), \tag{8}$$

where α is a learned scalar gate and MHA denotes multi-head attention. The resulting tokens F'_t represent the current observation after it has been enriched with recent history that indicates the active short-horizon continuation.

We also append the projected history-evidence summary as a single context token:

$$e_t^{\mathrm{tok}} = \mathrm{reshape}(e_t) \in \mathbb{R}^{1\times d}, \quad C_t = [F'_t; e_t^{\mathrm{tok}}]. \tag{9}$$

Conceptually, the condition-dependent information represented by C_t is the learned short-horizon intent representation $m_t = f_\phi(o_t, \ell, h_t^K)$ introduced in Section 4.1. In implementation, this representation is realized by the current tokens after gated history fusion together with the appended history-evidence token. Following the DiT-based conditional flow-matching action heads used in (Community, 2026; Intelligence et al., 2025; Bjorck et al., 2025), we use C_t as the conditioning context for chunk generation. At inference time, C_t is fixed for the current decision step, and the action chunk is obtained by starting from Gaussian noise and integrating the predicted conditional velocity field with the same Euler-style solver used in GR00T.

Training follows the standard conditional flow-matching objective. Given a target action chunk τ_t, Gaussian noise $\epsilon \sim \mathcal{N}(0, I)$, and a sampled flow time $s \sim p(s)$, we define the interpolated chunk $X_s = (1-s)\epsilon + s\tau_t$ and train the conditional velocity field $\hat{V}_\theta(X_s, s \mid C_t)$ to match the ground-truth displacement $\tau_t - \epsilon$:

$$\mathcal{L}_{\mathrm{flow}} = \mathbb{E}_{\mathcal{D},\epsilon,s}\left[\left\|\hat{V}_\theta(X_s, s \mid C_t) - (\tau_t - \epsilon)\right\|_2^2\right]. \tag{10}$$

Table 1: **Results on AliasBench.** We compare IntentVLA against Qwen3-VL-GR00T (Community, 2026), direct history-as-context baselines, and the memory-centric MemoryVLA baseline (Shi et al., 2026). 'OOM' means the corresponding training configuration runs out of GPU memory when past frames are fed directly into the Qwen backbone as extra context.

Method	Back Forth	Crossing Path	Bi-manual	Multi Goal	Avg
Qwen3-VL-GR00T (Community, 2026)	6.0	15.7	5.5	8.7	9.0
+ last 16 history frames	OOM	OOM	OOM	OOM	OOM
+ last 8 history frames	OOM	OOM	OOM	OOM	OOM
+ last 4 history frames	7.3	19.3	2.5	11.0	10.4
+ 4 frames uniformly sampled from last 16	31.8	47.3	6.0	18.7	28.1
MemoryVLA (Shi et al., 2026)	13.3	22.7	4.0	16.7	14.9
IntentVLA	**49.3**	**74.7**	**17.0**	**31.3**	**45.8**

5 Experiment

We begin with AliasBench, which directly tests the failure mode identified in the introduction. On this benchmark, we compare against Qwen3-VL-GR00T and several history-as-extra-context baselines that feed multiple past frames directly into the Qwen backbone. We then evaluate on SimplerEnv (Li et al., 2024c), LIBERO (Liu et al., 2023), RoboCasa-GR1 Tabletop Tasks (Nasiriany et al., 2024; Bjorck et al., 2025), and a real-world snack-cleanup task to test whether the same design transfers beyond the controlled ambiguity benchmark. Across all experiments, we focus on partially observed scenarios where one-frame conditioning is insufficient and analyze both success rate and rollout stability.

5.1 Results on AliasBench

For AliasBench, we sample 100 demonstrations per task and train all attempted methods under the same 30K-step, 16-H100 budget. Evaluation follows the default RoboTwin2 test protocol (Chen et al., 2025); task-specific maximum test steps are listed in Appendix E. Table 1 shows that direct raw-history conditioning is costly: the 8-frame and 16-frame variants run out of memory, and the best feasible variant reaches 28.1% average success. MemoryVLA (Shi et al., 2026) improves over the frame-only Qwen3-VL-GR00T baseline (14.9% vs. 9.0%), but remains below the sampled-history

Table 2: **Results of evaluating the VLA models with the WidowX robot in the SimplerEnv simulation environment (Li et al., 2024c).** We highlight the best results in **bold** and the second-best results with underline.

Method	Stack Block	Put Carrot	Put Spoon	Put Eggplant	Avg
RT-1-X (O'Neill et al., 2024)	0.0	4.2	0.0	0.0	1.1
Octo-Base (Team et al., 2024)	0.0	12.5	15.8	41.7	17.5
Octo-Small (Team et al., 2024)	0.0	8.2	41.7	56.7	26.7
OpenVLA-OFT (Kim et al., 2025)	30.0	30.0	34.2	72.5	41.8
RoboVLM (Li et al., 2024b)	0.0	37.5	50.0	83.3	42.7
Magma (Yang et al., 2025)	20.8	29.2	37.5	91.7	44.8
CogACT (Li et al., 2024a)	15.0	50.8	71.7	67.5	51.3
SpatialVLA (Qu et al., 2025)	25.0	20.8	20.8	70.8	34.4
TraceVLA (Zheng et al., 2025b)	16.6	16.6	12.5	65.0	27.7
VideoVLA (Shen et al., 2025)	45.8	20.8	75.0	70.8	53.1
π_0 (Black et al., 2024)	29.2	62.5	29.2	91.6	53.1
$\pi_{0.5}$ (Intelligence et al., 2025)	44.7	64.7	49.3	69.7	57.1
GR00T-N1.6 (GEAR-Team et al., 2025)	5.5	65.5	64.5	93.0	57.1
LangForce (Lian et al., 2026)	33.3	63.8	89.6	79.2	66.5
PhysBrain (Lin et al., 2025b)	34.7	58.3	90.3	80.6	65.9
3D-Mix (Yu et al., 2026a)	30.2	61.5	86.5	94.8	68.2
TwinBrainVLA (Yu et al., 2026b)	33.3	58.3	87.5	79.1	64.5
MemoryVLA (Shi et al., 2026)	37.5	75.0	75.0	100.0	71.9
Qwen3-VL-GR00T (Community, 2026)	18.8	59.4	83.0	100.0	65.3
IntentVLA	54.2	66.7	70.8	100.0	**72.9**

baseline. One possible reason is related to memory consolidation. MemoryVLA stores perceptual and cognitive entries in a memory bank and, when the number of entries exceeds the memory length, merges the most similar adjacent entries in each stream by averaging their vectors. This is effective for reducing redundant memories, but AliasBench is designed so that visually similar states can correspond to different short-horizon intents; Figure 3 shows that same-intent and different-intent neighbors can be extremely close in feature space. In this setting, similarity-based consolidation may average entries that are visually close but intent-distinct, making the resulting memory less discriminative for selecting the episode-specific continuation. IntentVLA reaches 45.8%, improving over Qwen3-VL-GR00T by 36.8 points and over the strongest feasible history-as-context baseline by 17.7 points. Gains are largest on crossing-path and back-and-forth tasks, where recent visual history reveals the object source or local phase. Bimanual and multi-goal tasks remain challenging; Appendix C further analyzes inter-chunk consistency.

5.2 Results on Standard Benchmarks

To evaluate whether the advantage of IntentVLA transfers beyond the controlled aliases in AliasBench, we further conduct experiments on the SimplerEnv, LIBERO, and RoboCasa. Appendix B provides the dataset and training details.

Results on SimplerEnv. For SimplerEnv (Li et al., 2024c), we evaluate on four WidowX manipulation tasks: *Put Spoon on Towel*, *Put Carrot on Plate*, *Stack Green Block on Yellow Block*, and *Put Eggplant in Yellow Basket*. The results are reported in Table 2. IntentVLA achieves the best overall average success rate of 72.9%, outperforming the Qwen3-VL-GR00T baseline by 7.6 points and exceeding the strongest previously reported average, 68.2% from 3D-Mix, by 4.7 points. The gains are especially large on *Put Carrot on Plate*, *Stack Green Block on Yellow Block*, and *Put Eggplant in Yellow Basket*. Although performance on *Put Spoon on Towel* is lower than the baseline, the overall result shows that recent visual history substantially improves robustness on partially observed chunked manipulation.

Table 3: **Comparison on the LIBERO benchmark.** We train one policy for all 4 suites.

Method	Spatial	Object	Goal	Long	Avg
OpenVLA (Kim et al., 2024)	87.4	88.4	79.2	53.7	76.5
OpenVLA-OFT (Kim et al., 2025)	97.6	98.4	97.9	94.5	97.1
π_0 (Black et al., 2024)	96.8	98.8	95.8	85.2	94.1
$\pi_{0.5}$ (Intelligence et al., 2025)	98.8	98.2	98.0	92.4	96.9
VLA-JEPA (Sun et al., 2026)	94.8	99.6	95.8	94.0	96.1
TwinBrainVLA (Yu et al., 2026b)	99.2	99.0	96.8	95.4	97.6
Qwen3-VL-GR00T (Community, 2026)	97.8	98.8	97.4	92.0	96.5
IntentVLA	99.3	99.7	98.1	97.4	**98.6**

Results on LIBERO. For LIBERO (Liu et al., 2023), we report Avg@500 success rates on Spatial, Object, Goal, and Long. As shown in Table 3, LIBERO is already close to saturation for strong recent VLAs, especially on the Spatial, Object, and Goal suites where several methods exceed 98%. Therefore, these suites leave limited room to diagnose the effect of short-horizon history. We focus on LIBERO-Long, which is the most relevant suite for our setting because it contains longer, multi-stage manipulation routines where adjacent local continuations must remain consistent. On LIBERO-Long, IntentVLA reaches 97.4%, compared with 92.0% for the Qwen3-VL-GR00T baseline and 92.4% for $\pi_{0.5}$. Although IntentVLA does not introduce an explicit long-horizon planner, recent visual history helps it infer the active local continuation inside a longer routine, which reduces inconsistent chunk generation across sub-steps.

Results on RoboCasa. We evaluate IntentVLA on the RoboCasa GR1 Tabletop Manipulation Benchmark (Nasiriany et al., 2024; Bjorck et al., 2025), which contains 24 diverse manipulation tasks with articulated objects and varied object geometries. The benchmark includes tasks such as *PnPBottleToCabinetClose* and *PnPCanToDrawer-*

Table 4: RoboCasa-GR1 simulation results on four representative pick-and-place tasks. The omitted columns indicate additional RoboCasa-GR1 tasks; Avg. is computed over all 24 tasks rather than only the shown tasks. Full per-task results are provided in Appendix Table 7.

Method	PnP Bottle	PnP Can	PnP Cup	PnP Milk	···	Avg.
GR00T N1.5 (Bjorck et al., 2025)	54.0	50.0	38.0	60.0	···	48.2
GR00T N1.6 (GEAR-Team et al., 2025)	51.5	13.0	8.5	14.0	···	47.6
VP-VLA (Wang et al., 2026)	54.0	72.0	44.0	74.0	···	53.8
TwinBrainVLA (Yu et al., 2026b)	74.0	72.0	52.0	60.0	···	54.6
PhysBrain (Lin et al., 2025b)	74.0	68.0	42.0	54.0	···	50.0
LangForce (Lian et al., 2026)	72.0	78.0	46.0	56.0	···	52.6
Full-Frame Training (Community, 2026)	46.0	80.0	54.0	48.0	···	47.8
IntentVLA (ours)	76.0	88.0	46.0	48.0	···	**57.0**

Close, as well as scenarios involving appliances such as microwaves and toasters. As shown in Table 4, IntentVLA achieves the best 24-task average success rate of 57.0%, outperforming TwinBrainVLA (54.6%), VP-VLA (53.8%), LangForce (52.6%), and Qwen3-VL-GR00T (47.8%). Full per-task results are provided in Appendix D. These results indicate that the short-horizon history signal remains useful even in a broader benchmark with articulated objects and varied interaction patterns.

5.3 Real-World Experiment

We evaluate IntentVLA on a real-world snack-cleanup task using two Franka Research 3 arms with Robotiq 85 grippers and Intel RealSense D435 head/hand cameras. Each episode starts with four randomly placed snacks, and the robot must place them into a bag within 500 action chunks. We train from 300 full four-snack demonstrations. The task creates multi-object ambiguity: several snacks and the bag remain visible, so a frame-only policy can start toward one snack but switch to another after replanning. Real camera noise, occlusion, and small execution deviations further test whether the policy preserves a stable target commitment. Table 5 compares IntentVLA with $\pi_{0.5}$ over 50 trials per model. IntentVLA reaches higher cumulative rates for collecting at least two and three snacks (62%/32% vs. 44%/18%), improves the expected count from 1.44 to 1.86, and is the only model to collect all four within budget.

5.4 Ablation Studies

Component ablations on SimplerEnv. Table 6 studies which components drive the gains on SimplerEnv. Removing VGGT and temporal history gives the frame-only Qwen3-VL-GR00T baseline. Adding VGGT to only the current frame does not improve the average score, showing that VGGT is not useful merely as another single-frame encoder; its benefit comes from geometric and inter-frame evidence in a recent window. History fusion raises the average from 65.3% to 69.5%, and the compact intent-evidence token further raises it to 72.9%, indicating that both temporal fusion and an explicit history summary help chunk generation.

The per-task results clarify where history helps. *Put Spoon on Towel* drops relative to the frame-only baseline because it has little short-horizon intent ambiguity: once the spoon and towel are visible, the current frame already carries most needed information, and history may dilute attention to the small, thin spoon. In contrast, *Stack Green Block on Yellow Block* improves from 18.8% to 54.2%, as stacking depends on relative 3D geometry, alignment, and the grasp-to-placement transition, where recent visual history and VGGT geometry tokens provide useful cues.

Table 5: **Real-world Experiment on a dual-arm Franka Research 3 setup.** Each rollout starts with four randomly placed snacks on the table and runs for a 500-action-chunk budget. Columns $\geq$1–$\geq$4 are cumulative threshold counts: the number of 50 trials in which at least that many snacks are placed into the bag, so they are not disjoint. $\mathbb{E}[\#]$ is the expected number of placed snacks per rollout.

Method	$\geq$1	$\geq$2	$\geq$3	$\geq$4	$\mathbb{E}[\#]$
$\pi_{0.5}$ (Intelligence et al., 2025)	41/50 (82%)	22/50 (44%)	9/50 (18%)	0/50 (0%)	1.44
IntentVLA (ours)	43/50 (86%)	31/50 (62%)	16/50 (32%)	3/50 (6%)	1.86

Table 6: **Ablation study on SimplerEnv.** We ablate the history encoder, temporal history, history fusion, and the compact intent-evidence token. Results are success rates on the four WidowX tasks.

Variant	Stack Block	Put Carrot	Put Spoon	Put Eggplant	Avg
Frame-only Qwen3-VL-GR00T	18.8	59.4	**83.0**	**100.0**	65.3
VGGT, current frame only	30.2	61.5	72.5	94.8	64.8
History fusion only, no intent token	49.0	65.6	67.7	95.8	69.5
IntentVLA	**54.2**	**66.7**	70.8	**100.0**	**72.9**

6 Conclusion

We introduced IntentVLA, a history-conditioned VLA for stabilizing chunk generation under partial observability, and AliasBench, a 12-task benchmark for short-horizon observation aliasing. Across AliasBench, standard simulation benchmarks, and a real-world cleanup task, compact recent-history conditioning improves success and inter-chunk consistency.

Limitations

IntentVLA focuses on recovering short-horizon intent from recent visual history. The history pathway is used only as an auxiliary signal for intent disambiguation during chunk generation; we do not claim general episodic memory or long-horizon memory as a contribution. Accordingly, our intended application scope is locally aliased manipulation, where recent frames reveal the active continuation but the current frame alone is ambiguous. Tasks that require remembering sparse events outside the recent window or recovering from large closed-loop deviations are outside this scope and may need longer-term memory or planning modules.

Ethical Considerations

This work studies short-horizon intent disambiguation for VLA policies in controlled simulation benchmarks and a small lab robot setup. It does not use personal, sensitive, or offensive language data. The real-world demonstrations consist of robot observations and actions collected by expert teleoperation, without identifying human information. The main ethical risk is misinterpretation: high benchmark scores should not be treated as evidence that a policy is safe or reliable for open-world physical deployment, and low scores are diagnostic signals under our aliasing setting, not a complete model characterization. We intend the benchmark, code, and model weights for research evaluation, with standard physical-robot safety practices, human supervision, and further robustness validation before deployment.

References

Hongzhe Bi, Lingxuan Wu, Tianwei Lin, Hengkai Tan, Zhizhong Su, Hang Su, and Jun Zhu. 2025. H-RDT: Human manipulation enhanced bimanual robotic manipulation. *arXiv preprint arXiv:2507.23523*.

Johan Bjorck, Fernando Castañeda, Nikita Cherniadev, Xingye Da, Runyu Ding, Linxi Fan, Yu Fang, Dieter Fox, Fengyuan Hu, Spencer Huang, and 1 others. 2025. GR00T N1: An open foundation model for generalist humanoid robots. *arXiv preprint arXiv:2503.14734*.

Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Lucy Xiaoyang Shi, and 5 others. 2024. π_0: A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*.

Qingwen Bu, Jisong Cai, Li Chen, Xiuqi Cui, Yan Ding, Siyuan Feng, Shenyuan Gao, Xindong He, Xu Huang, Shu Jiang, and 1 others. 2025a. Agibot world colosseo: A large-scale manipulation platform for scalable and intelligent embodied systems. *arXiv preprint arXiv:2503.06669*.

Qingwen Bu, Yanting Yang, Jisong Cai, Shenyuan Gao, Guanghui Ren, Maoqing Yao, Ping Luo, and Hongyang Li. 2025b. Univla: Learning to act anywhere with task-centric latent actions. *arXiv preprint arXiv:2505.06111*.

Tianxing Chen, Zanxin Chen, Baijun Chen, Zijian Cai, Yibin Liu, Zixuan Li, Qiwei Liang, Xianliang Lin, Yiheng Ge, Zhenyu Gu, and 1 others. 2025. Robotwin 2.0: A scalable data generator and benchmark with strong domain randomization for robust bimanual robotic manipulation. *arXiv preprint arXiv:2506.18088*.

Tianxing Chen, Yuran Wang, Mingleyang Li, Yan Qin, Hao Shi, Zixuan Li, Yifan Hu, Yingsheng Zhang, Kaixuan Wang, Yue Chen, and 1 others. 2026a. Rmbench: Memory-dependent robotic manipulation benchmark with insights into policy design. *arXiv preprint arXiv:2603.01229*.

Yi Chen, Yuying Ge, Hui Zhou, Mingyu Ding, Yixiao Ge, and Xihui Liu. 2026b. Dial: Decoupling intent and action via latent world modeling for end-to-end vla. *arXiv preprint arXiv:2603.29844*.

StarVLA Community. 2026. Starvla: A lego-like codebase for vision-language-action model developing. *arXiv preprint arXiv:2604.05014*.

GEAR-Team, Allison Azzolini, Johan Bjorck, Valts Blukis, Fernando Castañeda, Rahul Chand, and 1 others. 2025. Gr00t n1.6: An improved open foundation model for generalist humanoid robots. https://research.nvidia.com/labs/gear/gr00t-n1_6/.

Renming Huang, Chendong Zeng, Wenjing Tang, Jintian Cai, Cewu Lu, and Panpan Cai. 2026. Mimic intent, not just trajectories. *arXiv preprint arXiv:2602.08602*.

Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Manuel Y. Galliker, Dibya Ghosh, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Devin LeBlanc, and 17 others. 2025. $\pi_{0.5}$: a vision-language-action model with open-world generalization. *arXiv preprint arXiv:2504.16054*.

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

Moo Jin Kim, Chelsea Finn, and Percy Liang. 2025. Fine-tuning vision-language-action models: Optimizing speed and success. *arXiv preprint arXiv:2502.19645*.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan P Foster, Pannag R Sanketi, Quan Vuong, Thomas Kollar, Benjamin Burchfiel, Russ Tedrake, Dorsa Sadigh, Sergey Levine, Percy Liang, and Chelsea Finn. 2024. OpenVLA: An open-source vision-language-action model. In *Conference on Robot Learning (CoRL)*.

Chengmeng Li, Junjie Wen, Yaxin Peng, Yan Peng, and Yichen Zhu. 2026a. Pointvla: Injecting the 3d world into vision-language-action models. *IEEE Robotics and Automation Letters*, 11(3):2506–2513.

Chengyang Li, Kaiyi Xiong, Yuan Xu, Lei Qian, Yizhou Wang, and Wentao Zhu. 2026b. Gazevla: Learning human intention for robotic manipulation. *arXiv preprint arXiv:2604.22615*.

Fuhao Li, Wenxuan Song, Han Zhao, Jingbo Wang, Pengxiang Ding, Donglin Wang, Long Zeng, and Haoang Li. 2025a. Spatial forcing: Implicit spatial representation alignment for vision-language-action model. *arXiv preprint arXiv:2510.12276*.

Peiyan Li, Yixiang Chen, Hongtao Wu, Xiao Ma, Xiangnan Wu, Yan Huang, Liang Wang, Tao Kong, and Tieniu Tan. 2025b. Bridgevla: Input-output alignment for efficient 3d manipulation learning with vision-language models. In *Advances in neural information processing systems (NeurIPS)*.

Qixiu Li, Yaobo Liang, Zeyu Wang, Lin Luo, Xi Chen, Mozheng Liao, Fangyun Wei, Yu Deng, Sicheng Xu, Yizhong Zhang, and 1 others. 2024a. CogACT: A foundational vision-language-action model for synergizing cognition and action in robotic manipulation. *arXiv preprint arXiv:2411.19650*.

Xinghang Li, Peiyan Li, Minghuan Liu, Dong Wang, Jirong Liu, Bingyi Kang, Xiao Ma, Tao Kong, Hanbo Zhang, and Huaping Liu. 2024b. Towards generalist robot policies: What matters in building vision-language-action models. *arXiv preprint arXiv:2412.14058*.

Xuanlin Li, Kyle Hsu, Jiayuan Gu, Oier Mees, Karl Pertsch, Homer Rich Walke, Chuyuan Fu, Ishikaa Lunawat, Isabel Sieh, Sean Kirmani, Sergey Levine, Jiajun Wu, Chelsea Finn, Hao Su, Quan Vuong, and Ted Xiao. 2024c. SimplerEnv: Evaluating real-world robot manipulation policies in simulation. In *Conference on Robot Learning (CoRL)*.

Shijie Lian, Bin Yu, Xiaopeng Lin, Laurence T Yang, Zhaolong Shen, Changti Wu, Yuzhuo Miao, Cong Huang, and Kai Chen. 2026. Langforce: Bayesian decomposition of vision language action models via latent action queries. *arXiv e-prints*, pages arXiv–2601.

Tao Lin, Gen Li, Yilei Zhong, Yanwen Zou, Yuxin Du, Jiting Liu, Encheng Gu, and Bo Zhao. 2025a. Evo-0: Vision-language-action model with implicit spatial understanding. *arXiv preprint arXiv:2507.00416*.

Xiaopeng Lin, Shijie Lian, Bin Yu, Ruoqi Yang, Changti Wu, Yuzhuo Miao, Yurun Jin, Yukun Shi, Cong Huang, Bojun Cheng, and 1 others. 2025b. PhysBrain: Human egocentric data as a bridge from vision language models to physical intelligence. *arXiv preprint arXiv:2512.16793*.

Yixin Lin, Austin S. Wang, Giovanni Sutanto, Akshara Rai, and Franziska Meier. 2021. Polymetis. https://facebookresearch.github.io/fairo/polymetis/.

Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. 2023. LIBERO: Benchmarking knowledge transfer for lifelong robot learning. *Advances in neural information processing systems (NeurIPS)*, 36:44776–44791.

Songming Liu, Bangguo Li, Kai Ma, Lingxuan Wu, Hengkai Tan, Xiao Ouyang, Hang Su, and Jun Zhu. 2026. Rdt2: Exploring the scaling limit of umi data towards zero-shot cross-embodiment generalization. *arXiv preprint arXiv:2602.03310*.

Songming Liu, Lingxuan Wu, Bangguo Li, Hengkai Tan, Huayu Chen, Zhengyi Wang, Ke Xu, Hang Su, and Jun Zhu. 2025. RDT-1b: a diffusion foundation model for bimanual manipulation. In *International Conference on Learning Representations (ICLR)*.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. In *International Conference on Learning Representations (ICLR)*.

Yulin Luo, Hao Chen, Zhuangzhe Wu, Bowen Sui, Jiaming Liu, Chenyang Gu, Zhuoyang Liu, Qiuxuan Feng, Jiale Yu, Shuo Gu, and 1 others. 2026. Look before acting: Enhancing vision foundation representations for vision-language-action models. *arXiv preprint arXiv:2603.15618*.

Soroush Nasiriany, Abhiram Maddukuri, Lance Zhang, Adeet Parikh, Aaron Lo, Abhishek Joshi, Ajay Mandlekar, and Yuke Zhu. 2024. RoboCasa: Large-scale simulation of everyday tasks for generalist robots. In *Robotics: Science and Systems*.

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

Karl Pertsch, Kyle Stachowicz, Brian Ichter, Danny Driess, Suraj Nair, Quan Vuong, Oier Mees, Chelsea Finn, and Sergey Levine. 2025. FAST: Efficient action tokenization for vision-language-action models. *arXiv preprint arXiv:2501.09747*.

Delin Qu, Haoming Song, Qizhi Chen, Yuanqi Yao, Xinyi Ye, Yan Ding, Zhigang Wang, JiaYuan Gu, Bin Zhao, Dong Wang, and 1 others. 2025. Spatialvla: Exploring spatial representations for visual-language-action model. *arXiv preprint arXiv:2501.15830*.

Jeff Rasley, Samyam Rajbhandari, Olatunji Ruwase, and Yuxiong He. 2020. Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 3505–3506.

Yichao Shen, Fangyun Wei, Zhiying Du, Yaobo Liang, Yan Lu, Jiaolong Yang, Nanning Zheng, and Baining Guo. 2025. VideoVLA: Video generators can be generalizable robot manipulators. In *Advances in neural information processing systems (NeurIPS)*.

Hao Shi, Bin Xie, Yingfei Liu, Lin Sun, Fengrong Liu, Tiancai Wang, Erjin Zhou, Haoqiang Fan, Xiangyu Zhang, and Gao Huang. 2026. Memoryvla: Perceptual-cognitive memory in vision-language-action models for robotic manipulation. In *International Conference on Learning Representations (ICLR)*.

Jingwen Sun, Wenyao Zhang, Zekun Qi, Shaojie Ren, Zezhi Liu, Hanxin Zhu, Guangzhong Sun, Xin Jin, and Zhibo Chen. 2026. Vla-jepa: Enhancing vision-language-action model with latent world model. *arXiv preprint arXiv:2602.10098*.

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

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, and 1 others. 2023. Bridgedata v2: A dataset for robot learning at scale. In *Conference on Robot Learning (CoRL)*, pages 1723–1736. PMLR.

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. 2025. Vggt: Visual geometry grounded transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Zixuan Wang, Yuxin Chen, Yuqi Liu, Jinhui Ye, Pengguang Chen, Changsheng Lu, Shu Liu, and Jiaya Jia. 2026. Vp-vla: Visual prompting as an interface for vision-language-action models. *arXiv preprint arXiv:2603.22003*.

Jianwei Yang, Reuben Tan, Qianhui Wu, Ruijie Zheng, Baolin Peng, Yongyuan Liang, Yu Gu, Mu Cai, Seonghyeon Ye, Joel Jang, and 1 others. 2025. Magma: A foundation model for multimodal ai agents. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14203–14214.

Bin Yu, Shijie Lian, Xiaopeng Lin, Zhaolong Shen, Yuliang Wei, Haishan Liu, Changti Wu, Hang Yuan, Bailing Wang, Cong Huang, and 1 others. 2026a. 3d-mix for vla: A plug-and-play module for integrating vggt-based 3d information into vision-language-action models. *arXiv preprint arXiv:2603.24393*.

Bin Yu, Shijie Lian, Xiaopeng Lin, Yuliang Wei, Zhaolong Shen, Changti Wu, Yuzhuo Miao, Xinming Wang, Bailing Wang, Cong Huang, and 1 others. 2026b. Twinbrainvla: Unleashing the potential of generalist vlms for embodied tasks via asymmetric mixture-of-transformers. *arXiv preprint arXiv:2601.14133*.

Xuanran Zhai, Qianyou Zhao, Qiaojun Yu, and Ce Hao. 2025a. Vfp: Variational flow-matching policy for multi-modal robot manipulation. *arXiv preprint arXiv:2508.01622*.

Xuanran Zhai, Qianyou Zhao, Qiaojun Yu, and Ce Hao. 2025b. Vfp: Variational flow-matching policy for multi-modal robot manipulation. *arXiv preprint arXiv:2508.01622*.

Haoyu Zhen, Xiaowen Qiu, Peihao Chen, Jincheng Yang, Xin Yan, Yilun Du, Yining Hong, and Chuang Gan. 2024. 3D-VLA: A 3D vision-language-action generative world model. In *International conference on machine learning (ICML)*, pages 61229–61245.

Jinliang Zheng, Jianxiong Li, Zhihao Wang, Dongxiu Liu, Xirui Kang, Yuchun Feng, Yinan Zheng, Jiayin Zou, Yilun Chen, Jia Zeng, and 1 others. 2025a. X-VLA: Soft-prompted transformer as scalable cross-embodiment vision-language-action model. *arXiv preprint arXiv:2510.10274*.

Ruijie Zheng, Yongyuan Liang, Shuaiyi Huang, Jianfeng Gao, Hal Daumé III, Andrey Kolobov, Furong Huang, and Jianwei Yang. 2025b. TraceVLA: Visual trace prompting enhances spatial-temporal awareness for generalist robotic policies. *arXiv preprint arXiv:2412.10345*.

Linqing Zhong, Yi Liu, Yifei Wei, Ziyu Xiong, Maoqing Yao, Si Liu, and Guanghui Ren. 2026. Acot-vla: Action chain-of-thought for vision-language-action models. *arXiv preprint arXiv:2601.11404*.

Zheyuan Zhou, Liang Du, Zixun Sun, Xiaoyu Zhou, Ruimin Ye, Qihao Chen, Yinda Chen, and Lemiao Qiu. 2026. Main-vla: Modeling abstraction of intention and environment for vision-language-action models. *arXiv preprint arXiv:2602.02212*.

Brianna Zitkovich, Tianhe Yu, Sichun Xu, Peng Xu, Ted Xiao, Fei Xia, Jialin Wu, Paul Wohlhart, Stefan Welker, Ayzaan Wahid, and 1 others. 2023. Rt-2: Vision-language-action models transfer web knowledge to robotic control. In *Conference on Robot Learning (CoRL)*, pages 2165–2183.

A Reproducibility

To support reproducibility, we provide the benchmark generation code and model implementation in the supplementary Software archive. We also provide video examples from the sampled benchmark datasets in the supplementary Data archive.

B Training and Data Details

Unless otherwise specified, the standard-benchmark experiments are built on the StarVLA training pipeline (Community, 2026) and run on 16 NVIDIA H100 GPUs. We follow the default StarVLA training protocol for fair comparison, and use AdamW (Loshchilov and Hutter, 2017) with an initial learning rate of 1×10^{-5} and a cosine annealing schedule. System-level optimizations include DeepSpeed ZeRO-2 (Rasley et al., 2020), gradient clipping with maximum norm 1.0, and no gradient accumulation.

SimplerEnv. We use the Bridge-DataV2 (Walke et al., 2023) subset of Open X-Embodiment (O'Neill et al., 2024) and fine-tune the model for 30K steps on 16 GPUs with batch size 16 per device. We evaluate the policy with the official SimplerEnv evaluation scripts (Li et al., 2024c).

LIBERO. We train a single policy jointly across the four LIBERO suites and evaluate Spatial, Object, Goal, and Long.

RoboCasa-GR1. For RoboCasa-GR1, we use the Humanoid Robot Tabletop Manipulation subset of PhysicalAI Robotics-GR00T-X-Embodiment-Sim (Bjorck et al., 2025) and fine-tune the model for 30K steps on 16 GPUs with batch size 16 per device.

C Additional Analysis on Intent Consistency and Mode Switching

C.1 Inter-Chunk Consistency in Ambiguous Windows

We further evaluate whether the policy preserves the same local intent across adjacent replanning steps. Consider a chunk $\hat{\tau}^{(t)} = (\hat{a}_t^{(t)}, \ldots, \hat{a}_{t+H-1}^{(t)})$ generated at decision time t, and another chunk $\hat{\tau}^{(t+r)}$ generated after replanning at time $t + r$. These two chunks overlap on future time steps $t + r, \ldots, t + H - 1$. We define the inter-chunk consistency error with an L2 action distance:

$$\mathrm{ICC}_t = \frac{1}{H - r} \sum_{j=r}^{H-1} \left\| \hat{a}_{t+j}^{(t)} - \hat{a}_{t+j}^{(t+r)} \right\|_2^2. \quad (11)$$

Here $\hat{a}_{t+j}^{(t)}$ denotes the action for absolute time $t + j$ predicted by the chunk sampled at time t, while $\hat{a}_{t+j}^{(t+r)}$ denotes the prediction for the same absolute time made by the next replanning step. We refer to this metric as ICC-L2. It is computed only inside annotated ambiguity windows in Alias-Bench, where the current observation alone does not identify the correct continuation. Lower ICC-L2 is better: it means that adjacent chunks agree more strongly on their overlapping future segment, which is the expected action-level signature of preserved short-horizon intent.

Figure 5 shows that IntentVLA reduces inter-chunk inconsistency across all 12 tasks. Averaged over tasks, mean ICC-L2 decreases from 0.219 to 0.181, a 17.6% relative reduction. The family-level view also shows lower tail inconsistency across all ambiguity families. These results indicate that recent visual history makes adjacent replanned action chunks more consistent in the ambiguous regions where frame-conditioned chunk policies are likely to change intent.

C.2 Mode Switching Under Receding-Horizon Sampling

The failure mode studied in this paper is not that a policy can represent multiple valid behaviors. The problem is that, under an aliased current observation, the policy may remain *uncommitted* about which local continuation is active. Receding-horizon execution then turns this uncommitted multimodality into a temporal consistency problem: two adjacent action chunks may each be plausible in isolation, but they can correspond to different short-horizon intents.

To make this point explicit, suppose the policy replans every r environment steps. Let $p_t(z)$ and $p_{t+r}(z)$ denote conceptual intent distributions at two adjacent decision steps, where $z \in \mathcal{Z}$ indexes local continuations such as task phase, source-conditioned destination, or handoff direction. These distributions are only used for analysis; IntentVLA does not explicitly infer a discrete intent label. If the two chunks are sampled independently from these intent distributions, the probability that

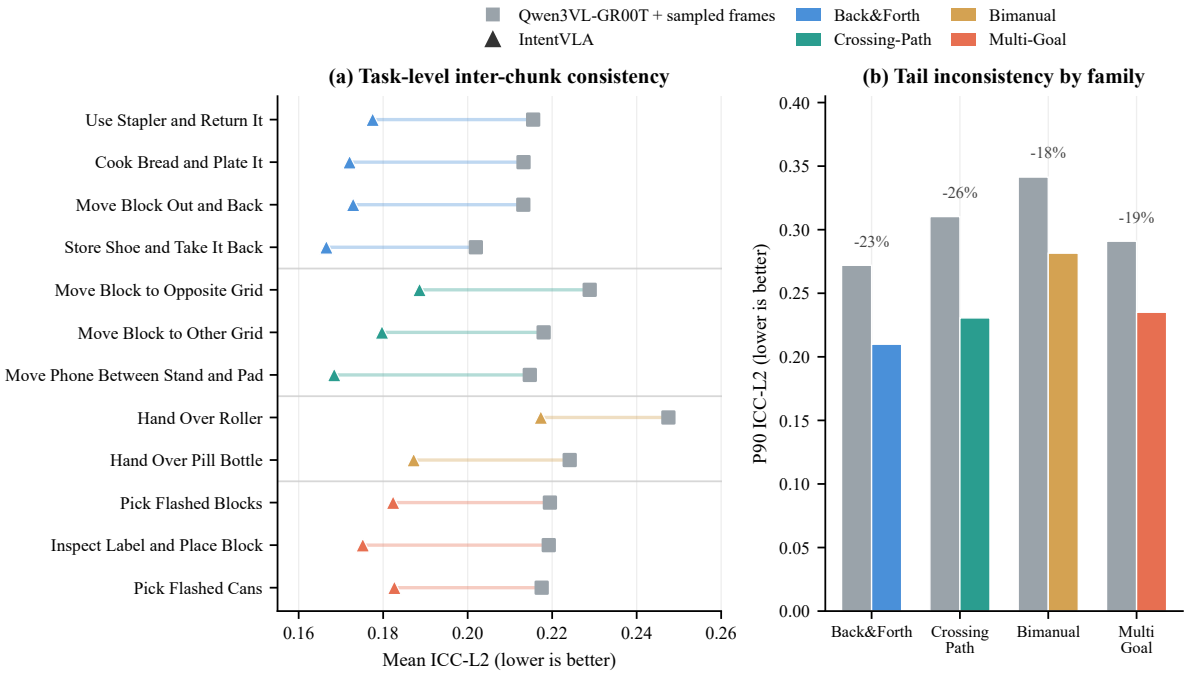

Figure 5: **Inter-chunk consistency in AliasBench ambiguity windows.** We compare IntentVLA against the strongest feasible history-as-context baseline in Table 1, Qwen3-VL-GR00T with four frames uniformly sampled from the last 16 frames. ICC-L2 is the squared L2 overlap error defined in Eq. (11); lower values indicate more consistent local intent. Left: task-level mean ICC-L2. Right: family-level 90th-percentile ICC-L2, where the 90th percentile measures tail inconsistency among the harder ambiguity windows. In the right panel, gray bars show the results of Qwen3-VL-GR00T with sampled frames.

they correspond to different intents is

$$P_{\text{switch}}(t, r) = 1 - \sum_{z \in \mathcal{Z}} p_t(z)\, p_{t+r}(z). \qquad (12)$$

This quantity is a diagnostic, not an additional training objective. In an aliased region, the current frame may leave several continuations plausible. If $p_t(z) = p_{t+r}(z)$ is uniform over M plausible intents, then $P_{\text{switch}}(t, r) = 1 - 1/M$: adjacent chunks are likely to switch intent even though neither chunk is individually invalid. By contrast, if recent history concentrates both distributions around the same committed intent $z^\star$, then the switch probability approaches zero. Thus, the role of history is not to remove multimodality across episodes, but to preserve within-episode commitment during local replanning.

The latent intent z is not observed for a learned policy, so we evaluate the consequence of mode switching in action space. When two adjacent chunks predict actions for overlapping absolute timesteps, a switch in short-horizon intent should appear as disagreement between the two predictions. The ICC-L2 metric in Eq. (11) measures exactly this overlap disagreement inside annotated ambiguity windows. A large ICC-L2 therefore indicates a possible action-level manifestation of mode switching, while a small ICC-L2 indicates that adjacent chunks remain aligned with the same continuation. This makes ICC-L2 an observable proxy for the consistency effect implied by $P_{\text{switch}}(t, r)$.

The full ICC statistics in Figure 5 support this interpretation. Beyond the mean reduction reported in the main text, the task-averaged 90th-percentile ICC-L2 decreases from 0.298 to 0.233, a 21.7% relative reduction, and the standard deviation across ambiguity windows drops from 0.093 to 0.046. This indicates that IntentVLA reduces both average overlap disagreement and unstable high-error windows. At the family level, the 90th-percentile ICC-L2 improves consistently across back-and-forth, crossing-path, bimanual, and multi-goal ambiguity, with relative reductions of 22.8%, 25.7%, 17.5%, and 19.2%, respectively. These results match the mode-switching analysis: recent visual history makes the effective short-horizon intent conditioning more committed, and adjacent sampled chunks become less likely to follow different continuations in aliased states.

D Full RoboCasa-GR1 Results

Table 7 reports the full 24-task RoboCasa-GR1 results corresponding to the average scores in Table 4.

E AliasBench Task Definitions

AliasBench is designed to isolate short-horizon observation aliasing rather than generic long-horizon memory demands. All 12 tasks follow the same core principle: the current frame alone is insufficient to determine the correct continuation, but the missing information is still available in the recent episode context. We summarize the benchmark families in Table 9 and then list the task-level definitions used in our evaluation.

Following the default RoboTwin2 evaluation protocol (Chen et al., 2025), each rollout terminates either when the task succeeds or when it reaches a task-specific maximum number of environment steps. Since the RoboTwin2 report specifies trial-based evaluation but does not prescribe a single global maximum horizon for our newly sampled tasks, we set each maximum test budget to twice the corresponding training sampling horizon. The resulting budgets are listed in Table 8.

Benchmark families and visual examples. The four families are intended to capture common manipulation patterns rather than synthetic edge cases. **Back-and-forth ambiguity** covers repeated local routines in which nearly identical carrying or staging states reappear in different phases. **Crossing-path ambiguity** covers source-dependent routing, where similar transport states arise from different recent origins. **Bimanual ambiguity** captures dual-arm settings in which center or handoff configurations can look nearly symmetric. **Multi-goal ambiguity** covers scenes with multiple plausible objects or destinations, where a transient cue or recently revealed property selects the active target.

Figure 2 provides several visual examples from AliasBench. In *Move Phone Between Stand and Pad*, a natural everyday command is something like "hey, put the phone on the other stand." However, in the third frame, when the robot arm is already holding the phone in mid-air, once one action chunk has finished and the policy must generate the next chunk, the current observation alone no longer reveals which phone stand is the starting point and which one is the target. In *Cook Bread and Plate It*, the first frame, where the robot picks up the bread, and the fifth frame, where it puts the bread down, look similar; likewise, the second frame, where the bread is placed onto the skillet for cooking, and the fourth frame, where it is taken back out and moved for plating, are also visually similar, even though they correspond to different intents. Similarly, in the third frame of *Hand Over Roller*, transferring the roller from left to right and transferring it from right to left can produce similar observations, but the subsequent intents are fundamentally different. These are exactly the kinds of short-horizon aliases that a frame-conditioned chunk policy cannot reliably resolve from the current frame alone.

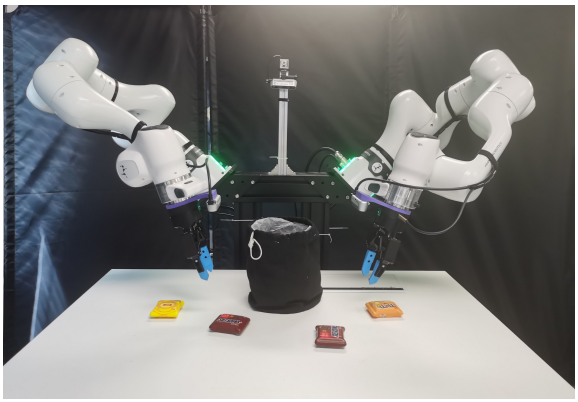

Figure 6: Dual-arm Franka setup for snack cleanup.

F Real-world robot setup.

Figure 6 shows the real-world snack-cleanup setup: two Franka Research 3 arms with Robotiq 85 grippers. Intel RealSense D435 cameras capture observations from head and end-effector views. We use Polymetis (Lin et al., 2021) for teleoperation and collect human expert demonstrations.

Table 7: **Full RoboCasa-GR1 Tabletop results.** We report Avg@50 success rates (%) for 24 tasks. Results for Isaac-GR00T N1.5 and N1.6 are sourced from the official Isaac-GR00T repository (Bjorck et al., 2025); Qwen3-VL-GR00T results are sourced from the official StarVLA experiments (Community, 2026).

Task	GR00T N1.5	GR00T N1.6	Qwen3-VL GR00T	VP VLA	TwinBrain VLA	Phys Brain	Lang Force	IntentVLA
PnP Bottle To Cabinet Close	54.0	51.5	46.0	54.0	74.0	74.0	72.0	76.0
PnP Can To Drawer Close	50.0	13.0	80.0	72.0	72.0	68.0	78.0	88.0
PnP Cup To Drawer Close	38.0	8.5	54.0	44.0	52.0	42.0	46.0	46.0
PnP Milk To Microwave Close	60.0	14.0	48.0	74.0	60.0	54.0	56.0	48.0
PnP Potato To Microwave Close	32.0	41.5	28.0	34.0	36.0	24.0	36.0	44.0
PnP Wine To Cabinet Close	38.0	16.5	46.0	48.0	46.0	54.0	46.0	56.0
PnP * to * Close (Avg)	45.3	24.2	50.3	54.3	56.7	52.7	55.7	**59.7**
PnP Novel From Cuttingboard To Basket	38.0	58.0	48.0	66.0	62.0	62.0	66.0	66.0
PnP Novel From Cuttingboard To Cardboardbox	46.0	46.5	40.0	54.0	46.0	44.0	40.0	52.0
PnP Novel From Cuttingboard To Pan	58.0	68.5	68.0	74.0	70.0	56.0	68.0	56.0
PnP Novel From Cuttingboard To Pot	62.0	65.0	52.0	54.0	66.0	58.0	48.0	54.0
PnP Novel From Cuttingboard To Tieredbasket	28.0	46.5	56.0	56.0	52.0	40.0	44.0	46.0
PnP Novel From Cuttingboard To * (Avg)	46.4	56.9	52.8	**60.8**	59.2	52.0	53.2	54.8
PnP Novel From Placemat To Basket	30.0	58.5	42.0	48.0	30.0	42.0	54.0	56.0
PnP Novel From Placemat To Bowl	60.0	57.5	44.0	74.0	54.0	56.0	62.0	76.0
PnP Novel From Placemat To Plate	56.0	63.0	48.0	70.0	64.0	80.0	52.0	58.0
PnP Novel From Placemat To Tieredshelf	36.0	28.5	18.0	26.0	38.0	14.0	24.0	32.0
PnP Novel From Placemat To * (Avg)	45.5	51.9	38.0	54.5	46.5	48.0	48.0	**55.5**
PnP Novel From Tray To Cardboardbox	52.0	51.5	38.0	44.0	46.0	40.0	50.0	52.0
PnP Novel From Tray To Plate	48.0	71.0	56.0	66.0	72.0	66.0	58.0	68.0
PnP Novel From Tray To Pot	60.0	64.5	50.0	38.0	56.0	52.0	62.0	66.0
PnP Novel From Tray To Tieredbasket	52.0	57.0	36.0	58.0	46.0	50.0	44.0	42.0
PnP Novel From Tray To Tieredshelf	32.0	31.5	16.0	24.0	28.0	22.0	22.0	20.0
PnP Novel From Tray To * (Avg)	48.8	**55.1**	39.2	46.0	49.6	46.0	47.2	49.6
PnP Novel From Plate To Bowl	58.0	57.0	60.0	52.0	60.0	54.0	54.0	60.0
PnP Novel From Plate To Cardboardbox	44.0	43.5	50.0	44.0	46.0	50.0	48.0	64.0
PnP Novel From Plate To Pan	60.0	51.0	54.0	56.0	56.0	68.0	54.0	66.0
PnP Novel From Plate To Plate	64.0	78.7	70.0	62.0	66.0	78.0	78.0	76.0
PnP Novel From Plate To * (Avg)	56.5	57.6	58.5	53.5	57.0	62.5	58.5	**66.5**
Average	48.2	47.6	47.8	53.8	54.6	50.0	52.6	**57.0**

Family	**Task**	**Max. Test Steps**
Back-and-forth	Move Block Out and Back	900
Back-and-forth	Cook Bread and Plate It	1100
Back-and-forth	Use Stapler and Return It	900
Back-and-forth	Store Shoe and Take It Back	900
Crossing-path	Move Block to the Opposite Grid	500
Crossing-path	Move Block to the Other Grid	500
Crossing-path	Move Phone Between Stand and Pad	700
Bimanual	Hand Over Roller	900
Bimanual	Hand Over Pill Bottle	900
Multi-goal	Pick Flashed Blocks in Order	1500
Multi-goal	Pick Flashed Cans in Order	1200
Multi-goal	Inspect Label and Place Block	700

Table 8: **AliasBench evaluation budgets.** We follow the default RoboTwin2 test protocol and terminate each rollout at task success or at the task-specific maximum number of environment steps. Each maximum test budget is set to twice the corresponding training sampling horizon.

Family	# Tasks	Core latent factor	Definition
Back-and-forth ambiguity	4	Current local phase	Repeated local states recur in different phases of a short routine, but the required next action changes with phase.
Crossing-path ambiguity	3	Recent source / origin	Similar transport states arise after different pickup origins, and the correct target depends on that recent source.
Bimanual ambiguity	2	Handoff direction / side of origin	Similar center or handoff states arise in dual-arm transfer, but the next hand and continuation depend on transfer direction.
Multi-goal ambiguity	3	Transient cue / hidden property	Multiple local candidates coexist, and a brief cue or recently observed property selects the correct target.

Table 9: Overview of the four task families in AliasBench.

Task	Task definition	Core latent factor	Why the current frame is insufficient
Move Block Out and Back	One arm moves a red block from grid A to grid B and then back to A.	Current phase	Similar carrying states appear in both halves, but the next placement differs between the outbound and return phases.
Cook Bread and Plate It	A bread slice is moved from a plate to a skillet and then back to the plate.	Current phase	Holding the bread near the workspace can mean placing it on the skillet or returning it to the plate.
Use Stapler and Return It	A stapler is moved from pad A to pad B and then back from B to A.	Current phase	Similar transport states recur across two short phases, but the next destination changes with phase.
Store Shoe and Take It Back	A shoe is placed into a shoebox and later moved back out to an external target area.	Current phase	Holding the shoe near the box can mean inserting it or taking it back out.

Table 10: Back-and-forth ambiguity tasks in AliasBench.

Task	Task definition	Core latent factor	Why the current frame is insufficient
Move Block to the Other Grid	A red block starts on one of two grids and must be moved to the other grid.	Recent source	Once the block is in hand, the transport state looks similar for both start grids, but the correct destination is the opposite source grid.
Move Block to the Opposite Grid	A red block starts on one of four grids in a 2×2 layout and must be moved to the diagonally opposite grid.	Recent source	Similar mid-transport states occur for different start grids, but the correct diagonal target depends on the pickup origin.
Move Phone Between Stand and Pad	A phone begins either on a flat area or on a stand and must be moved to the other location.	Recent source	Carrying the phone looks similar in both cases, while the correct target depends on whether it came from the stand or the flat area.

Table 11: Crossing-path ambiguity tasks in AliasBench.

Task	Task definition	Core latent factor	Why the current frame is insufficient
Hand Over Pill Bottle	A pill bottle starts on the left or right outer pad, moves to the center pad, and is then sent to the opposite outer pad by the other hand.	Handoff continuation	At the center pad, both hands and the bottle can look similar, but the correct next hand depends on which side the bottle came from.
Hand Over Roller	A long roller starts on one side, is moved to the centerline, handed over, and then placed on the opposite side.	Handoff direction	Near the centerline, the handoff state is nearly symmetric, yet the continuation depends on transfer direction.

Table 12: Bimanual ambiguity tasks in AliasBench.

Task	Task definition	Core latent factor	Why the current frame is insufficient
Pick Flashed Blocks in Order	Three blocks are available, and the object to be picked briefly flashes underneath before being placed onto the current target pad.	Active-target cue	After the flash disappears, several candidate blocks remain, but the frame no longer reveals which one is active.
Pick Flashed Cans in Order	Two cans are presented, and the can to be picked briefly flashes underneath before being placed onto the current target pad.	Active-target cue	Once the flash vanishes, multiple candidate picks remain plausible in the current frame.
Inspect Label and Place Block	The robot inspects a hidden label on a block and then places it onto the matching target area instead of a distractor area.	Hidden property	During final transport and placement, the label may no longer be visible, so the target depends on the recent inspection result.

Table 13: Multi-goal ambiguity tasks in AliasBench.