

Keep the Future, Drop the Rollout: RIFT for World Action Models

Chushan Zhang¹ Jinguang Tong¹ Xuesong Li¹ Yikai Wang^{3,*} Hongdong Li^{1,*}

¹Australian National University ²Beijing Normal University

*Corresponding authors

Abstract

World action models (WAMs) condition robot actions on predicted futures, but iterative video rollout increases deployment latency. We ask whether action generation requires the evolving rollout trajectory or only its future representation. Across four WAMs on all 40 LIBERO tasks, paired closed-loop interventions show that masking or reassigning future-cache values changes execution and reduces success, indicating sensitivity to future values and their assigned positions. For Joint and Cosmos-2, however, replaying one fixed final-clean key/value (K/V) cache nearly preserves unmodified execution, with 1.7 to 1.9 cm end-effector average displacement error and 97.9% to 98.2% success. This separates cache consumption from production: these models can reuse a fixed cache but still require iterative rollout to construct it. We therefore propose RIFT (*Rollout-free Imagination via Future Tokens*), which uses learned anticipation tokens to construct a complete future K/V cache in one backbone pass while retaining the original future-read interface. On LIBERO, RIFT achieves 98.8% success, close to rollout-based Joint, IDM, and LingBot-VA at 98.4% to 98.6%, while reducing action-chunk latency by 68.2% to 89.1%. On RoboTwin 2.0, RIFT reaches 92.9/92.6% on clean/randomized scenes, the highest observed among the evaluated methods. These results support rollout-free future conditioning without iterative video generation at deployment.

1 Introduction

World action models (WAMs) couple future video prediction with robot control. A single network predicts a short video and conditions its next actions on that prediction (Yuan et al. 2026; Li et al. 2026; Bi et al. 2025). In matched settings, policies that retain this future read achieve higher success than current-only variants. However, iterative video generation makes rollout-based systems incur $3.3\times$ to $9.6\times$ the latency of current-only deployment (fig. 1).

Existing efficient variants remove the future read at deployment. Fast-WAM drops the future branch after future-prediction co-training (Yuan et al. 2026), whereas PFD distills a future-conditioned correction into the current-only path (Fang, Chen, and Cai 2026). Both reduce latency but retain a gap to policies that explicitly read a generated future. Yet the action expert consumes a future representation, whereas iterative rollout is the process that constructs it. Existing comparisons change both the availability of this

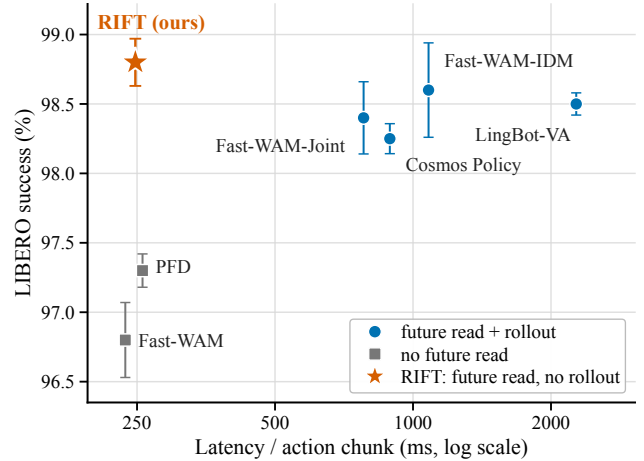


Figure 1: **LIBERO success versus deployment latency.** Points show mean success and bars show $\pm$ std over three evaluation seeds of one checkpoint per method (2,000 trials per seed); latency is ms per action chunk on one A800. Gray squares use no future read; blue circles read a future representation produced by rollout; RIFT (orange star) reads one-pass anticipation tokens without rollout. The x-axis is logarithmic; the y-axis is truncated. Methods use their original denoising configurations.

representation and the process used to construct it, so their separate contributions remain unresolved. We therefore ask whether a WAM can preserve explicit test-time future conditioning without iterative video rollout.

To separate these factors, we examine the future-position K/V cache, the per-layer channel from predicted futures to action tokens. Joint co-denoising updates this cache at every denoising step, whereas the generate-then-act inverse dynamics model (IDM) exposes only a final clean cache. Their similar success suggests that the representation, rather than its iterative trajectory, may provide the shared benefit. Success alone cannot reveal what the action reads.

We therefore intervene on this cache. The attention mask prevents video tokens from attending to action tokens, so the cache is an action-independent intervention site. We record it and replay action denoising with non-target inputs

fixed, either masking the future read or editing future values under the recorded keys. Each closed-loop intervention uses the same initial state and policy seed as the unmodified policy. We measure success rate (SR) and end-effector average displacement error (EE-ADE), the average drift from the unmodified end-effector trajectory.

Across 2,000 paired trials on all 40 LIBERO tasks, masking the future read yields 18.7 cm EE-ADE and reduces success from 98.4% to 9.7%. Spatial permutation and temporal swapping yield similar EE-ADE (14.3 and 15.6 cm) but sharply different success (65.2% and 0.7%), showing that the action reads future content at its assigned positions. In contrast, replaying final clean values under the original keys yields only 1.9 cm EE-ADE and 97.9% success. Under the original keys, the action therefore depends strongly on future value content and its organization, but little on how the future values evolve across denoising steps.

These findings provide a value-side constraint rather than a complete cache recipe. They do not show that the key trajectory can be frozen or that a complete cache can be produced without rollout. This limitation motivates the stronger hypothesis that one pass can produce the complete future interface. We propose *RIFT (Rollout-free Imagination via Future Tokens)*, which places learned anticipation tokens at future temporal positions and fills their per-layer K/V cache with one video-backbone pass. The action expert consumes this cache through the original future-read interface. We shape the anticipation states with conditional flow matching, using a distributional objective rather than direct L2 regression to the single observed future. Deployment requires one cache prefill followed by the ordinary action flow, without video diffusion or video decoding at test time.

On LIBERO, *RIFT* achieves 98.8% success, compared with 96.8% for current-only Fast-WAM and within the same success tier as rollout-based Joint (98.4%) and IDM (98.6%). It runs at $1.1 \times$ current-only latency rather than $3.3 \times$ to $9.6 \times$ for rollout-based alternatives. On RoboTwin 2.0, *RIFT* reaches 92.9/92.6% success on clean/randomized scenes, the highest observed among the evaluated methods. These results indicate that, for the studied WAM family, explicit future-position representations can support rollout-level success without requiring iterative video generation at deployment.

We make three contributions.

1. **We introduce a paired closed-loop intervention protocol for future caches.** The protocol records the action-independent cache, then either masks the future read or edits its values under the recorded keys before measuring the resulting physical execution with EE-ADE and SR.
2. **We identify two properties of the future cache.** Actions are highly sensitive to removing the future read or reassigning values across space and time, while final clean values under the original keys nearly preserve execution.
3. **We design and validate a rollout-free future interface.** *RIFT* produces a complete future-position K/V cache in one backbone pass, matches the rollout-based success tier at $1.1 \times$ current-only latency, and reaches the highest observed LIBERO and RoboTwin 2.0 success.

2 Related Work

Five lines of work set up the question this paper asks: policies with current-only deployment, policies that render a future at deployment, policies that use future prediction only during training, runtime uncertainty monitors, and the interpretability methods we borrow from.

Vision-language-action policies. Vision-language-action policies map observations and language directly to actions through pretrained vision-language backbones (Zitkovich et al. 2023; Kim et al. 2024; Black et al. 2024a; Physical Intelligence et al. 2025; Bjorck et al. 2025; Shukor et al. 2025; Gemini Robotics Team et al. 2025), often paired with diffusion or flow-matching action heads (Chi et al. 2023; Liu et al. 2024; Wen et al. 2025). Because the mapping is direct, inference needs no iterative video branch, which makes these policies the natural latency reference for methods that add one. Their action experts receive no explicit future-position state. Our current-only baseline occupies this regime, and the gap it leaves is what we try to close without rollout cost.

World action models. A complementary line makes future prediction explicit. Classical world-model methods learn dynamics and use them for planning or control (Schrittwieser et al. 2020; Hafner et al. 2023; Hansen, Su, and Wang 2024). More recent robot policies condition on generated future video (Du et al. 2023; Black et al. 2024b; Bharadhwaj et al. 2024; Zhou et al. 2024) or jointly learn future and action representations (Wu et al. 2024; Cheang et al. 2024; Hu et al. 2025; Li et al. 2025; Zhu et al. 2025; Feng et al. 2025; Pai et al. 2025; Won et al. 2025; Liang et al. 2025; Jang et al. 2025; Zhao et al. 2025; Cen et al. 2025; Kim et al. 2026; Liao et al. 2025). World action models tighten this coupling so that a single network both imagines and acts (Ye et al. 2026; Yuan et al. 2026; Li et al. 2026; Bi et al. 2025). Two interfaces dominate. Imagine-then-execute systems generate a future video and then apply inverse dynamics to it, so the action reads a fixed clean representation. Mixture-of-transformers systems denoise video and action through shared attention, so the action reads a representation that changes at every denoising step. These two designs expose the future in almost opposite ways yet report similar success. Our intervention study starts from this observation. Both pay for iterative video generation, which dominates their latency. Prior comparisons report end-task success only, which cannot distinguish a policy that uses the imagined future from one that merely benefits from training alongside it.

Implicit-future policies. A third line keeps future prediction as training-time signal while avoiding a visual rollout at inference. Fast-WAM removes the explicit future representation at deployment and preserves much of the aggregate success through a world-aware current representation, which shows how much of the benefit survives co-training alone (Yuan et al. 2026). PFD instead distills the action-side effect of a generated future into a lightweight current-only correction (Fang, Chen, and Cai 2026; Vapnik and Vashist 2009; Chen et al. 2019). FLARE (Zheng et al. 2025) and DreamVLA (Zhang et al. 2025) learn future-prediction tokens through auxiliary objectives without an explicit test-

time rollout, and Being-H0.7 trains latent queries with a future-informed posterior branch that is discarded at inference (Luo et al. 2026). EvoScene-VLA supervises an action-updated recurrent scene prefix with future scene-token targets and likewise discards its training-time scene predictor at deployment (Zhang et al. 2026). These methods remove a rollout-produced future read, distill its effect, or use future targets to train a compact latent state. In our matched evaluation, Fast-WAM and PFD leave a residual gap to rollout-based policies. RIFT takes the opposite decomposition: it keeps an explicit test-time future-position interface and replaces the iterative producer with a single learned pass.

Flow-matching uncertainty and runtime monitoring. Concurrent work reads uncertainty from the action flow itself: Rao et al. (2026) measures denoising-path acceleration along a single FM action trajectory, validates it against the L2 divergence of resampled action chunks, and accumulates the score with CUSUM for failure detection. Our auxiliary readouts expose a complementary signal in predicted future-latent space. In an optional shadow mode, the stopped-gradient L2 probe supplies a deterministic readout while the conditional-FM head supplies sampled modes; their cross-head discrepancy, normalized by within-FM spread, measures cross-estimator conflict rather than action-flow curvature. Writing μ for the probe readout and $\bar{x}$ for the mean of K FM samples x_i , the score is $d_{\text{ratio}} = \|\mu - \bar{x}\|_2^2 / (K^{-1} \sum_i \|x_i - \bar{x}\|_2^2 + \epsilon)$. Thus their proxy estimates uncertainty internal to one FM action head, whereas ours asks whether two future estimators agree. Both signals can miss confidently wrong predictions. This monitor is not part of the policy-only deployment or latency results.

Locating computation by intervention. Our instrument follows causal tracing, which intervenes on selected internal activations to measure their causal effect (Meng et al. 2022). We apply this idea to a robot policy’s attention cache rather than to an autoregressive language model’s hidden states. This matters because the standard alternatives answer a weaker question: linear probes establish that information is decodable from a representation, not that downstream computation uses it, and attention weights are similarly unreliable as evidence of use (Alain and Bengio 2016; Belinkov 2022; Jain and Wallace 2019; Serrano and Smith 2019). Intervening on the future read and measuring the executed trajectory tests use directly. Two properties of the WAM setting make this test unusually clean. The video-to-action attention mask makes the recorded cache action-independent, so we can either mask the read or edit its values under fixed keys. Closed-loop execution then supplies a physical readout in centimeters rather than a distance in an arbitrary latent space. For value edits, the remaining caveat is distributional rather than positional, and we return to it in the next section.

3 What does the action expert read?

3.1 The channel we edit

With paired closed-loop interventions, we test whether action needs the future read, whether its values must stay at assigned positions, and whether the complete K/V cache must evolve

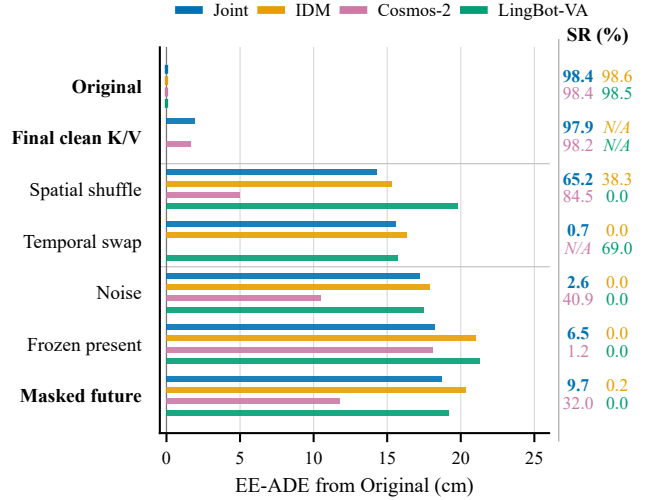


Figure 2: Future-cache interventions alter executed trajectories and task success. Value-edit rows retain the recorded Original key trajectory and modify only future-position values. Final-clean replay instead substitutes one final-clean K/V cache at every action-denoising step, while masking removes the future read. Bar length gives task-macro EE-ADE in centimeters; the compact block at right gives SR in percent. Each reported intervention result uses 2,000 paired trials over all 40 LIBERO tasks and is paired with the same model’s Original. Original has zero EE-ADE by definition and is marked by colored ticks at the origin. All numbers are direct measurements. Missing bars are not zero: IDM-style models already read one fixed final-clean future K/V cache, while Cosmos-2 exposes only one future timestep and therefore admits no temporal swap.

during denoising. Each model–intervention estimate uses 2,000 paired trials across all 40 LIBERO tasks. Figure 2 reports executed EE-ADE and success rate.

Fast-WAM-Joint, Fast-WAM-IDM, Cosmos Policy, and LingBot-VA construct futures differently but expose the same per-layer video K/V interface at future positions. We call it the *future cache*; *Original* denotes the unmodified checkpoint. We mask the read, edit future values under recorded keys, or replace K and V with one final-clean cache.

Because video tokens cannot attend to action tokens, the cache is action-independent given observation (o), language (l), and video-generation randomness. This property supports record and replay. The record pass generates video normally and stores per-layer future K/V at every action-denoising step. Value corruptions retain the recorded key trajectory and edit only the matched future values. The final-clean control instead takes the final clean future from Original’s iterative generation, prefills its complete K/V once, and reuses that fixed cache at every action-denoising step. All non-target inputs remain fixed. Exact replay of the unedited trajectory reproduces Original.

Shuffle and noise are location-exact but out of distribution, so we interpret them only beside the structured frozen-

present and final-clean K/V controls.

3.2 Scoring each edit

Action chunks are not directly comparable across architectures, so we execute paired policies. For episode i , intervention I and Original share the initial state and policy seed. Let $\mathbf{x}_{i,t}^I$ and $\mathbf{x}_{i,t}^O$ denote their recorded end-effector positions after environment step t . EE-ADE averages their distance over the common executed prefix T_i :

$$\text{EE-ADE}_i(I) = \frac{1}{T_i} \sum_{t=1}^{T_i} \|\mathbf{x}_{i,t}^I - \mathbf{x}_{i,t}^O\|_2. \quad (1)$$

We exclude the reset pose and use recorded simulator positions rather than integrating predicted actions. When runs end at different times, only their common prefix contributes. We average episode scores within each task before macro-averaging tasks; section B defines the task-cluster bootstrap. EE-ADE measures drift from Original; success measures task completion. High drift with similar success indicates another route; high drift with low success associates the intervention with failure during closed-loop task execution.

3.3 Intervention set

The interventions map directly to the three questions. Masking removes the read; norm-matched noise replaces its values; and frozen-present supplies plausible, non-predictive values. Spatial shuffle permutes values within frames, temporal swap exchanges value frames, and final-clean replay replaces the evolving complete cache with the same final-clean K/V at every action step. We apply each supported edit to Fast-WAM-Joint, Fast-WAM-IDM, Cosmos Policy, and LingBot-VA, comparing each with its Original. Claims are therefore within-model; cross-model magnitudes remain descriptive because architectures and checkpoints differ.

3.4 Finding 1: WAM action experts use future values at their assigned positions

Across all four WAMs, masking yields 11.8–20.4 cm EE-ADE and reduces success from 98.4–98.6% to 0.0–32.0%. Under recorded keys, noise yields 10.5–17.9 cm and at most 40.9% success, while frozen-present values yield 18.1–21.3 cm and at most 6.5%. These within-model effects show that action experts use meaningful future values.

Position also matters: spatial shuffle yields 5.0–19.8 cm and 0.0–84.5% success across all four, while temporal swap yields 15.6–16.3 cm and 0.0–69.0% on Joint, IDM, and LingBot-VA. Every supported edit changes execution and lowers own-model success, so future values are not an unordered pool. Severity differs by interface: temporal swap is worse for Joint and IDM, spatial shuffle for LingBot-VA, and Cosmos-2 exposes no temporal swap. Thus, position sensitivity is shared, not a universal severity ordering.

3.5 Finding 2: one final-clean K/V cache nearly preserves execution

Finding 1 establishes content and position sensitivity, not whether the complete cache must evolve. Where supported,

final-clean replay replaces the entire future-cache trajectory with one final-clean cache, holding both K and V fixed at every action-denoising step.

For Joint and Cosmos-2, final-clean K/V replay gives 1.9/1.7 cm EE-ADE and 97.9/98.2% success; their Originals reach 98.4/98.4%. These are the smallest nonzero EE-ADEs among compatible edits. IDM and LingBot-VA already expose one fixed final-clean K/V cache, so this replay is identical by construction and structurally N/A.

This result establishes consumption-side sufficiency: once final-clean K/V is available, Joint and Cosmos-2 nearly preserve execution without the evolving cache trajectory. It does not show that this cache can be produced without rollout, because its clean future came from iterative video generation. The full-cache intervention also does not isolate the separate contributions of keys and values. Section 4 tests the distinct producer-side hypothesis that one learned prefill can construct an effective fixed, complete K/V interface.

4 RIFT: One-pass Future-Token Imagination

4.1 From findings to our design

The analysis shows that WAM action experts require meaningful, position-bound future values and that Joint and Cosmos-2 can reuse one final-clean, complete K/V cache throughout action denoising with near-Original execution. This establishes a fixed complete cache as a viable consumption interface, but not its rollout-free production: the intervention cache still comes from iterative video generation. Thus, RIFT tests whether one learned prefill can produce the complete interface.

4.2 Writing the cache in one pass

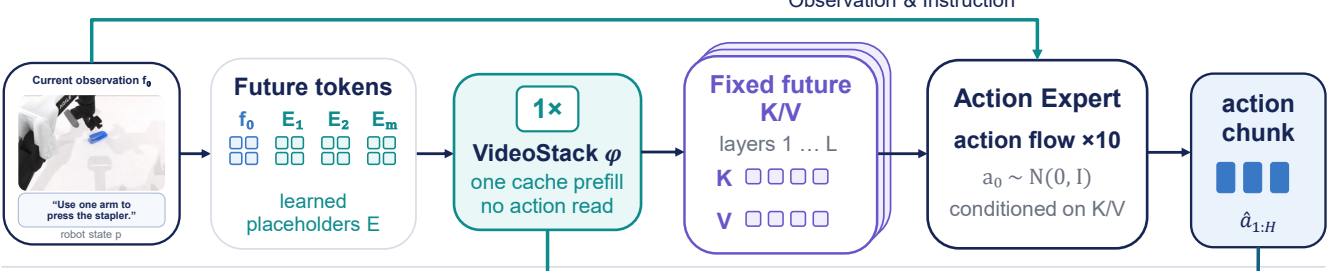
RIFT keeps Fast-WAM-Joint’s architecture and future-read interface. For a video stack with hidden width d and L layers, it replaces rolled-out future tokens with learned anticipation tokens $E \in \mathbb{R}^{m \times d}$. Each token inherits its corresponding future spatiotemporal index. If each latent frame contains n tokens and the clip contains T_{lat} latent frames, full alignment uses $m = n(T_{\text{lat}} - 1)$. As fig. 3 shows, we use full alignment with $m = 196$ on LIBERO and $m = 240$ on RoboTwin 2.0.

Let $f_0(o)$ denote the first-frame tokens extracted from observation o ; the remaining observation and instruction l enter through the backbone’s original conditioning path. Let ϕ collect the shared video expert’s parameters and the learned tokens E ; together they form the cache producer. One video-stack prefill writes the full future-position cache:

$$\begin{aligned} C_\phi(o, l) &= \left\{ \left(K_E^{(\ell)}, V_E^{(\ell)} \right) \right\}_{\ell=1}^L \\ &= \text{CachePrefill}_\phi([f_0(o); E], o, l), \end{aligned} \quad (2)$$

where $K_E^{(\ell)}, V_E^{(\ell)} \in \mathbb{R}^{m \times d}$ denote the layer- ℓ keys and values of the m anticipation tokens. We retain Fast-WAM-Joint’s mask: first-frame tokens attend only the observed frame, anticipation tokens attend the first frame and one another, and video tokens never attend action tokens. The cache is therefore action-independent.

(a) One-pass future read



(b) Training objectives

Native video co-training

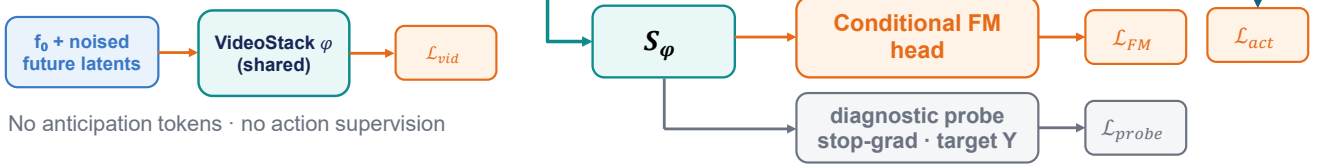


Figure 3: **RIFT training and deployment.** (a) One VideoStack prefill maps the first-frame latent and anticipation tokens to a fixed per-layer future-position K/V cache, which the action expert reuses throughout denoising. (b) Training pairs native video supervision with a deployment-matched forward for action, conditional-FM, and a stopped-gradient mean-squared probe loss. Action rows use the clean first frame; late perturbation affects only future-supervised rows. At policy-only test time, video co-training, auxiliary heads, and ground-truth futures are removed, leaving the prefill, fixed cache, and action flow.

The action expert reads $\mathcal{C}_\phi$ through the rollout model’s per-layer future-position interface:

$$\hat{a}_{1:H} = \text{ActionDenoise}(o, l; \mathcal{C}_\phi(o, l)). \quad (3)$$

Here, H is the action-chunk horizon; we use $H = 32$. $\mathcal{C}_\phi$ has no action-flow index: the same K/V serve every denoising evaluation, matching the fixed-cache consumption pattern tested in Finding 2. The producer-side hypothesis is that one pass from (o, l) constructs a usable cache; deployment then needs one prefill per chunk, with no rollout or VAE decoding.

4.3 Training

Training uses two forwards through the same video expert per optimization step. Deployment retains only the second forward’s cache prefill.

Native video supervision. The first forward applies native video-flow loss $\mathcal{L}_{\text{vid}}$ (Yuan et al. 2026) to a clean-first-frame clip with noised future latents and no anticipation tokens or action supervision. This preserves dynamics supervision while changing only the deployment interface.

Deployment-matched action training. The second forward matches deployment through input $[f_0; E]$ and its attention mask. Clean rows train the action expert on this cache with Fast-WAM’s inherited action flow-matching loss $\mathcal{L}_{\text{act}}$ (Yuan et al. 2026). After 70% of the configured curriculum horizon, the probability and scale of first-frame latent noise rise linearly from zero to 0.3 and 0.06 times the latent standard deviation. Perturbed rows retain $\mathcal{L}_{\text{FM}}$ and $\mathcal{L}_{\text{probe}}$ but

are masked out before $\mathcal{L}_{\text{act}}$ is reduced. Thus, $\mathcal{L}_{\text{act}}$ averages only clean rows, so all action-loss inputs are clean.

Conditional flow-matching supervision. Let $S_\phi \in \mathbb{R}^{m \times d}$ be the final anticipation states from the same deployment-matched forward and $Y \in \mathbb{R}^{m \times d_y}$ the aligned ground-truth future latent patches, where d_y is the dimension of one flattened latent patch. Under a multimodal conditional distribution, direct ℓ_2 regression has a conditional-mean optimum and can average distinct valid futures; we instead use conditional flow matching (Lipman et al. 2023) as a distributional auxiliary objective. We sample $\epsilon \sim \mathcal{N}(0, I)$ and $\sigma \in [0, 1]$ with the native video-flow schedule, then define

$$X_\sigma = (1 - \sigma)Y + \sigma\epsilon, \quad v_\sigma^* = \epsilon - Y. \quad (4)$$

Conditioned on S_ϕ , the training-only FM head v_ψ , parameterized by ψ , predicts the velocity from (X_σ, σ) . Using the native video-flow timestep weight $w_{\text{vid}}(\sigma)$, its loss is

$$\mathcal{L}_{\text{FM}} = \mathbb{E}_{Y, \epsilon, \sigma} \left[w_{\text{vid}}(\sigma) \|v_\psi(X_\sigma, \sigma; S_\phi) - v_\sigma^*\|_2^2 \right], \quad (5)$$

This auxiliary head shapes the cache producer during training but enters neither action nor policy-only deployment.

Stopped-gradient linear probe. The direct-L2 recipe provides a deterministic future-latent readout. In the final conditional-FM recipe, we retain this view as a linear probe, $\hat{Y}_{\text{L2}} = g_\omega(\text{RMS}(\text{stopgrad}(S_\phi)))$. We train it with MSE,

$$\mathcal{L}_{\text{probe}} = \text{mean}((\hat{Y}_{\text{L2}} - Y)^2).$$

The reduction weights every future-token and latent-channel squared residual uniformly before the batch mean. Detached input confines this loss to the probe; section A details the reduction. In optional shadow mode, the probe point prediction and conditional-FM samples define a normalized disagreement score. Both read the same anticipation states, but neither feeds the controller; section E defines the score.

Objective and gradient routes. Both forwards share the video expert. $\mathcal{L}_{\text{vid}}$ updates this expert and its video head; $\mathcal{L}_{\text{act}}$ updates the action expert and backpropagates through the cache into the shared video expert and E ; $\mathcal{L}_{\text{FM}}$ updates the shared video expert, E , and the FM head; and detached $\mathcal{L}_{\text{probe}}$ updates only the probe.

$$\mathcal{L} = \mathcal{L}_{\text{vid}} + \mathcal{L}_{\text{act}} + \lambda_{\text{FM}}\mathcal{L}_{\text{FM}} + \lambda_{\text{probe}}\mathcal{L}_{\text{probe}}, \quad (6)$$

Both auxiliary weights stay at 1 for the first 70% of the curriculum horizon, then follow a cosine decay to 0.2 over the final 30%. Policy-only deployment retains one $[f_0; E]$ prefill, fixed $\mathcal{C}_\phi$, and the standard action flow; section A gives remaining settings.

5 Experiments

5.1 Setup

Implementation. For LIBERO, Fast-WAM, Fast-WAM-Joint, Fast-WAM-IDM, and RIFT share the same Wan2.2-5B (Team Wan et al. 2025) Fast-WAM backbone, training data, and 20k-step budget, so differences come from the future interface rather than scale or data. We additionally compare with the Fast-WAM-based released PFD checkpoint (Fang, Chen, and Cai 2026) and the embodied-pretrained LingBot-VA (Li et al. 2026). Unless stated otherwise RIFT means the full conditional-FM recipe, with RIFT-L2 reserved for the base recipe in the ablations. Latency is milliseconds per action chunk on one A800 under each method’s own denoising configuration; RIFT’s figure covers its cache prefill and action denoising and excludes optional diagnostic readouts. In the tables, *future read* denotes explicit future-position attention; *rollout* denotes iterative future generation at deployment.

Benchmarks. LIBERO (Liu et al. 2023) has 40 tasks across its Spatial, Object, Goal, and Long suites. We evaluate one checkpoint per method with three evaluation seeds and 2,000 trials per seed (50 episodes per task), reporting mean and standard deviation across seeds; the error bars therefore measure closed-loop evaluation noise, not variation across training runs. For out-of-distribution (OOD) evaluation on LIBERO-Plus (Fei et al. 2026), we run Fast-WAM, Fast-WAM-Joint, Fast-WAM-IDM, and RIFT with one rollout on each of the benchmark’s 10,030 variants, without further training. RoboTwin 2.0 (Chen et al. 2025) has 50 bimanual tasks under clean and domain-randomized scenes. Following Yuan et al. (2026), the matched Fast-WAM/Joint/PFD/RIFT family trains on 2,500 clean-scene and 25,000 randomized demonstrations for 30k steps, with externally pretrained LingBot-VA for comparison. Each checkpoint is evaluated for 100 trials per task in each setting.

Table 1: **LIBERO success and deployment cost.** SR is mean $\pm$ std over three evaluation seeds of one checkpoint (2,000 trials each). Latency is ms per action chunk on one A800 under each method’s original denoising configuration. *Emb. PT.*: embodied pretraining. $\dagger$ released checkpoint evaluated by us; * our matched reproduction.

Method	Emb. PT.	Fut. read	Roll-out	SR (%)	Lat. (ms)	Rel.
Fast-WAM $\dagger$	×	×	×	96.8 $\pm$ 0.27	235.7	1.0×
PFD $\dagger$	×	×	×	97.3 $\pm$ 0.12	257.0	1.1×
Fast-WAM-Joint*	×	✓	✓	98.4 $\pm$ 0.26	780.2	3.3×
Fast-WAM-IDM*	×	✓	✓	98.6 $\pm$ 0.34	1081.2	4.6×
LingBot-VA $\dagger$	✓	✓	✓	98.5 $\pm$ 0.08	2270.3	9.6×
RIFT (ours)	×	✓	×	98.8 $\pm$ 0.17	247.9	1.1×

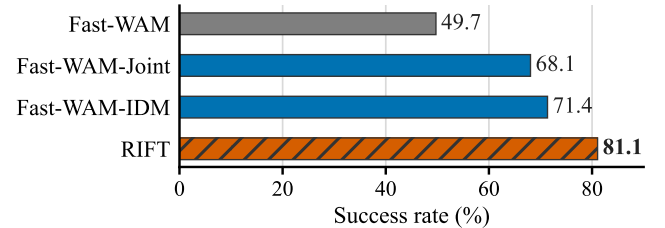


Figure 4: **Overall LIBERO-Plus OOD robustness.** Success across all 10,030 variants, with one rollout per variant and no further training; bars are point estimates.

5.2 Quantitative results

LIBERO. In Table 1, RIFT achieves 98.8% overall success, close to the 98.4% to 98.6% achieved by rollout-based Joint, IDM, and LingBot-VA. Unlike these methods, RIFT requires only 247.9 ms per action chunk, reducing latency by 68.2% to 89.1% while remaining close to current-only Fast-WAM at 235.7 ms. Compared with rollout-free Fast-WAM and PFD, RIFT improves success by 2.0 and 1.5 percentage points, respectively, at comparable latency.

LIBERO-Plus OOD. Across four checkpoints and 10,030 variants, RIFT achieves the highest overall success rate of 81.1%, a +9.7 percentage-point gain over Fast-WAM-IDM (fig. 4). This shows robustness to OOD perturbations. See table 5 for the full breakdown.

RoboTwin 2.0. The interface transfers to a second embodiment (table 2; per-task rates in table 6). RIFT reaches 92.9/92.6 on clean/randomized scenes, the best observed among the evaluated methods, against 92.5/92.1 for PFD, 92.4/91.4 for rollout-based LingBot-VA, 91.9/91.6 for Fast-WAM, and 91.0/91.1 for rollout-based Fast-WAM-Joint. We observe the same performance recovery on RoboTwin 2.0 while retaining the one-pass deployment path.

5.3 Ablations

Anticipation-token supervision. The base recipe RIFT-L2 regresses future latents with a direct L2 loss and reaches 98.37 $\pm$ 0.12; the conditional-FM recipe reaches 98.8 $\pm$ 0.17.

Table 2: **RoboTwin 2.0 closed-loop success (%)** on clean and domain-randomized scenes. Values aggregate one check-point per method; This report follows single-seed evaluation protocol. *Emb. PT.*: embodied pretraining.

Method	Emb. PT.	Fut. read	Roll-out	Clean	Rand.	AVG
LingBot-VA	✓	✓	✓	92.4	91.4	91.9
Fast-WAM-Joint	×	✓	✓	91.0	91.1	91.0
Fast-WAM	×	×	×	91.9	91.6	91.8
PFD	×	×	×	92.5	92.1	92.3
RIFT (ours)	×	✓	×	92.9	92.6	92.8

Both use the same one-pass graph and 247.9 ms cost, isolating supervision without deployment overhead. The 0.4 point difference approaches the evaluation’s resolution, and the full recipe includes its conditioning curriculum. Table 4 gives per-suite rates; we report FM as the recipe.

The number of anticipation tokens. Figure 5 sweeps $m = 2$ to full alignment ($m = 196$); current-only Fast-WAM is the no-cache $m = 0$ reference (96.75%). RIFT-L2 rises from 97.08% to 98.37%; conditional FM exceeds it from $m = 4$ and peaks at 98.78%. Even small interfaces beat the reference, and full alignment is best for both.

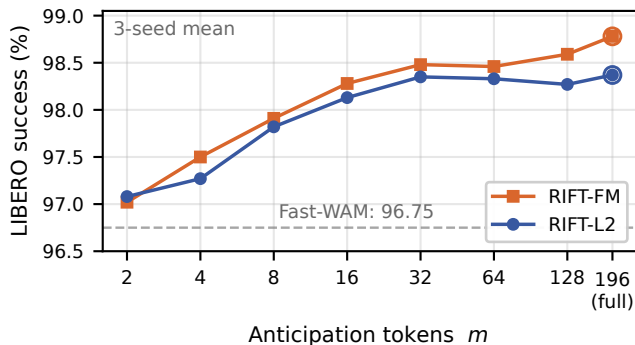


Figure 5: **Anticipation-interface capacity on LIBERO.** Mean success over four suites (three seeds) as token count m grows for L2 and conditional-FM supervision. The dashed line is Fast-WAM without anticipation; full alignment ($m = 196$) gives both recipes their best mean.

5.4 Qualitative results

Matched imagined futures. Figure 6 compares Fast-WAM-Joint rollout and RIFT one-pass decodes from matched starts on both benchmarks. Both evolve similarly at frames 0, 4, and 8. These visuals diagnose future representations, not cache equivalence; section G shows all frames.

L2-FM uncertainty warning. The stopped-gradient L2 probe and conditional-FM head yield controller-independent future estimates whose normalized disagreement defines a CUSUM warning (section E). Calibrated on 1,967 successful episodes, the mean CUSUM over 33 failed rollouts crosses η 210 steps before the common $t = 420$ endpoint (fig. 7).

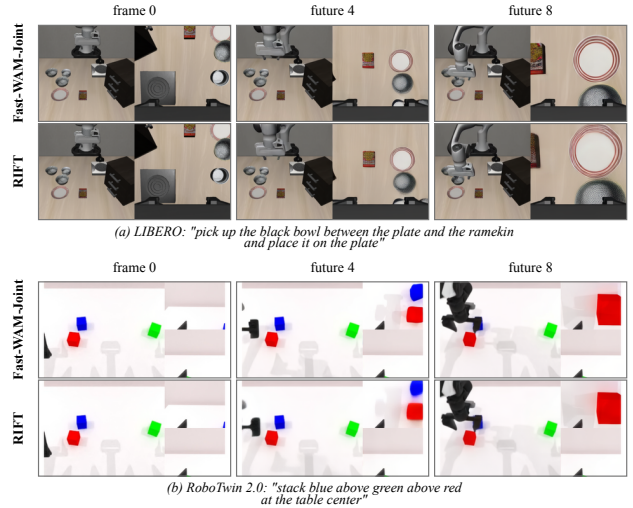


Figure 6: **Matched imagined futures.** From matched initial states, Fast-WAM-Joint iterative rollout (top) and RIFT one-pass anticipation decodes (bottom) show similar evolution at frames 0, 4, and 8 on LIBERO (upper block) and RoboTwin 2.0 (lower block). Decodes are diagnostic.

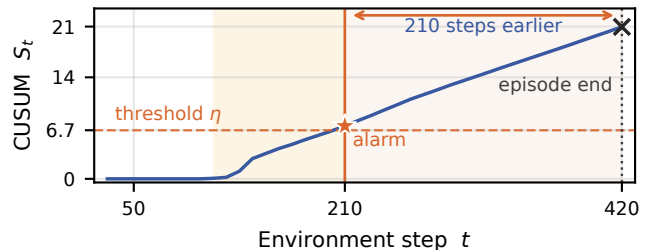


Figure 7: **L2-FM uncertainty warning over LIBERO.** Here η is the CUSUM alarm threshold, conformally calibrated on all successful tasks; failures are excluded from calibration. Across the 33 failed rollouts, the detector raises an alarm an average of 210 steps before failure.

6 Conclusion

World action models combine a representation read by the action expert with the iterative rollout that produces it. Intervening on the future K/V interface shows that actions require values bound to token positions, while one fixed final-clean K/V cache nearly reproduces Original execution within 1.9 cm EE-ADE. Because this cache remains rollout-produced, the intervention establishes consumption-side sufficiency rather than rollout-free production. RIFT addresses the remaining production problem with one anticipation-token prefill, preserving the complete future K/V interface while removing video denoising and VAE decoding. It clears the current-only gap on LIBERO at $1.1\times$ baseline latency. Without retraining, it also attains the highest overall performance among four evaluated checkpoints across all 10,030 LIBERO-Plus variants. It reaches the best observed RoboTwin 2.0 success in both evaluation settings.

References

- Alain, G.; and Bengio, Y. 2016. Understanding Intermediate Layers Using Linear Classifier Probes. *arXiv preprint arXiv:1610.01644*.
- Belinkov, Y. 2022. Probing Classifiers: Promises, Shortcomings, and Advances. *Computational Linguistics*, 48(1): 207–219.
- Bharadhwaj, H.; Dwibedi, D.; Gupta, A.; Tulsiani, S.; Doersch, C.; Xiao, T.; Shah, D.; Xia, F.; Sadigh, D.; and Kirmani, S. 2024. Gen2Act: Human Video Generation in Novel Scenarios Enables Generalizable Robot Manipulation. *arXiv preprint arXiv:2409.16283*.
- Bi, H.; Tan, H.; Xie, S.; Wang, Z.; Huang, S.; Liu, H.; Zhao, R.; Feng, Y.; Xiang, C.; Rong, Y.; Zhao, H.; Liu, H.; Su, Z.; Ma, L.; Su, H.; and Zhu, J. 2025. Motus: A Unified Latent Action World Model. *arXiv preprint arXiv:2512.13030*.
- Bjorck, J.; Castañeda, F.; Cherniadev, N.; Da, X.; Ding, R.; Fan, L.; Fang, Y.; Fox, D.; Hu, F.; Huang, S.; et al. 2025. GR00T N1: An Open Foundation Model for Generalist Humanoid Robots. *arXiv preprint arXiv:2503.14734*.
- Black, K.; Brown, N.; Driess, D.; Esmail, A.; Equi, M.; Finn, C.; Fusai, N.; Groom, L.; Hausman, K.; Ichter, B.; et al. 2024a. π_0 : A Vision-Language-Action Flow Model for General Robot Control. *arXiv preprint arXiv:2410.24164*.
- Black, K.; Nakamoto, M.; Atreya, P.; Walke, H.; Finn, C.; Kumar, A.; and Levine, S. 2024b. Zero-Shot Robotic Manipulation with Pre-Trained Image-Editing Diffusion Models. In *International Conference on Learning Representations (ICLR)*.
- Cen, J.; Huang, S.; Yuan, Y.; Li, K.; Yuan, H.; Yu, C.; Jiang, Y.; Guo, J.; Li, X.; Luo, H.; Wang, F.; Wang, F.; and Zhao, D. 2025. RynnVLA-002: A Unified Vision-Language-Action and World Model. *arXiv preprint arXiv:2511.17502*.
- Cheang, C.-L.; Chen, G.; Jing, Y.; Kong, T.; Li, H.; Li, Y.; Liu, Y.; Wu, H.; Xu, J.; Yang, Y.; Zhang, H.; and Zhu, M. 2024. GR-2: A Generative Video-Language-Action Model with Web-Scale Knowledge for Robot Manipulation. *arXiv preprint arXiv:2410.06158*.
- Chen, D.; Zhou, B.; Koltun, V.; and Krähenbühl, P. 2019. Learning by Cheating. In *Conference on Robot Learning (CoRL)*.
- Chen, T.; Chen, Z.; Chen, B.; Cai, Z.; Liu, Y.; Li, Z.; Liang, Q.; Lin, X.; Ge, Y.; Gu, Z.; et al. 2025. RoboTwin 2.0: A Scalable Data Generator and Benchmark with Strong Domain Randomization for Robust Bimanual Robotic Manipulation. *arXiv preprint arXiv:2506.18088*.
- Chi, C.; Feng, S.; Du, Y.; Xu, Z.; Cousineau, E.; Burchfiel, B.; and Song, S. 2023. Diffusion Policy: Visuomotor Policy Learning via Action Diffusion. In *Robotics: Science and Systems (RSS)*.
- Du, Y.; Yang, M.; Dai, B.; Dai, H.; Nachum, O.; Tenenbaum, J. B.; Schuurmans, D.; and Abbeel, P. 2023. Learning Universal Policies via Text-Guided Video Generation. In *Advances in Neural Information Processing Systems (NeurIPS)*.
- Fang, P.; Chen, H.; and Cai, X. 2026. Privileged Foresight Distillation: Zero-Cost Future Correction for World Action Models. *arXiv preprint arXiv:2604.25859*.
- Fei, S.; Wang, S.; Shi, J.; Dai, Z.; Cai, J.; Qian, P.; Ji, L.; He, X.; Zhang, S.; Fei, Z.; Fu, J.; Gong, J.; and Qiu, X. 2026. LIBERO-Plus: A Progressive Robustness Benchmark for Visual-Language-Action Models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 38574–38583.
- Feng, Y.; Tan, H.; Mao, X.; Xiang, C.; Liu, G.; Huang, S.; Su, H.; and Zhu, J. 2025. Vidar: Embodied Video Diffusion Model for Generalist Manipulation. *arXiv preprint arXiv:2507.12898*.
- Gemini Robotics Team; Abeyruwan, S.; Ainslie, J.; Alayrac, J.-B.; Arenas, M. G.; Armstrong, T.; Balakrishna, A.; Baruch, R.; Bauza, M.; Blokzijl, M.; et al. 2025. Gemini Robotics: Bringing AI into the Physical World. *arXiv preprint arXiv:2503.20020*.
- Hafner, D.; Pasukonis, J.; Ba, J.; and Lillicrap, T. 2023. Mastering Diverse Domains through World Models. *arXiv preprint arXiv:2301.04104*.
- Hansen, N.; Su, H.; and Wang, X. 2024. TD-MPC2: Scalable, Robust World Models for Continuous Control. In *International Conference on Learning Representations (ICLR)*.
- Hu, Y.; Guo, Y.; Wang, P.; Chen, X.; Wang, Y.-J.; Zhang, J.; Sreenath, K.; Lu, C.; and Chen, J. 2025. Video Prediction Policy: A Generalist Robot Policy with Predictive Visual Representations. In *International Conference on Machine Learning (ICML)*.
- Jain, S.; and Wallace, B. C. 2019. Attention is not Explanation. In *Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*.
- Jang, J.; Ye, S.; Lin, Z.; Xiang, J.; Bjorck, J.; Fang, Y.; Hu, F.; Huang, S.; Kundalia, K.; Lin, Y.-C.; et al. 2025. DreamGen: Unlocking Generalization in Robot Learning through Video World Models. *arXiv preprint arXiv:2505.12705*.
- Kim, M. J.; Gao, Y.; Lin, T.-Y.; Lin, Y.-C.; Ge, Y.; Lam, G.; Liang, P.; Song, S.; Liu, M.-Y.; Finn, C.; and Gu, J. 2026. Cosmos Policy: Fine-Tuning Video Models for Visuomotor Control and Planning. *arXiv preprint arXiv:2601.16163*.
- Kim, M. J.; Pertsch, K.; Karamcheti, S.; Xiao, T.; Balakrishna, A.; Nair, S.; Rafailov, R.; Foster, E.; Lam, G.; Sanketi, P.; Vuong, Q.; Kollar, T.; Burchfiel, B.; Tedrake, R.; Sadigh, D.; Levine, S.; Liang, P.; and Finn, C. 2024. OpenVLA: An Open-Source Vision-Language-Action Model. In *Conference on Robot Learning (CoRL)*.
- Li, L.; Zhang, Q.; Luo, Y.; Yang, S.; Wang, R.; Han, F.; Yu, M.; Gao, Z.; Xue, N.; Zhu, X.; Shen, Y.; and Xu, Y. 2026. Causal World Modeling for Robot Control. *arXiv preprint arXiv:2601.21998*.
- Li, S.; Gao, Y.; Sadigh, D.; and Song, S. 2025. Unified Video Action Model. *arXiv preprint arXiv:2503.00200*.
- Liang, J.; Tokmakov, P.; Liu, R.; Sudhakar, S.; Shah, P.; Ambrus, R.; and Vondrick, C. 2025. Video Generators Are Robot Policies. *arXiv preprint arXiv:2508.00795*.
- Liao, Y.; Zhou, P.; Huang, S.; Yang, D.; Chen, S.; Jiang, Y.; Hu, Y.; Cai, J.; Liu, S.; Luo, J.; Chen, L.; Yan, S.; Yao, M.; and Ren, G. 2025. Genie Envisioner: A Unified World Foundation Platform for Robotic Manipulation. *arXiv preprint arXiv:2508.05635*.

- Lipman, Y.; Chen, R. T. Q.; Ben-Hamu, H.; Nickel, M.; and Le, M. 2023. Flow Matching for Generative Modeling. In *International Conference on Learning Representations (ICLR)*.
- Liu, B.; Zhu, Y.; Gao, C.; Feng, Y.; Liu, Q.; Zhu, Y.; and Stone, P. 2023. LIBERO: Benchmarking Knowledge Transfer for Lifelong Robot Learning. In *Advances in Neural Information Processing Systems (NeurIPS)*.
- Liu, S.; Wu, L.; Li, B.; Tan, H.; Chen, H.; Wang, Z.; Xu, K.; Su, H.; and Zhu, J. 2024. RDT-1B: A Diffusion Foundation Model for Bimanual Manipulation. *arXiv preprint arXiv:2410.07864*.
- Luo, H.; Zhang, W.; Feng, Y.; Zheng, S.; Xu, H.; Xu, C.; Xi, Z.; Fu, Y.; and Lu, Z. 2026. Being-H0.7: A Latent World-Action Model from Egocentric Videos. *arXiv preprint arXiv:2605.00078*.
- Meng, K.; Bau, D.; Andonian, A.; and Belinkov, Y. 2022. Locating and Editing Factual Associations in GPT. In *Advances in Neural Information Processing Systems*, volume 35, 17359–17372.
- Pai, J.; Achenbach, L.; Montesinos, V.; Forrai, B.; Mees, O.; and Nava, E. 2025. mimic-video: Video-Action Models for Generalizable Robot Control beyond VLAs. *arXiv preprint arXiv:2512.15692*.
- Physical Intelligence; Black, K.; Brown, N.; Darpinian, J.; Dhabalia, K.; Driess, D.; Esmail, A.; Equi, M.; Finn, C.; Fusai, N.; et al. 2025. $\pi_{0.5}$: A Vision-Language-Action Model with Open-World Generalization. *arXiv preprint arXiv:2504.16054*.
- Rao, Z.; Zhao, Y.; Guo, W.; Fei, B.; Guo, Y.; and Xiong, H. 2026. The Geometry of Flow-Matching Uncertainty: A Cost-Free Uncertainty Proxy and Its Application in Flow-Based VLA Failure Detection. *arXiv preprint arXiv:2607.27933*.
- Schrittwieser, J.; Antonoglou, I.; Hubert, T.; Simonyan, K.; Sifre, L.; Schmitt, S.; Guez, A.; Lockhart, E.; Hassabis, D.; Graepel, T.; Lillicrap, T.; and Silver, D. 2020. Mastering Atari, Go, Chess and Shogi by Planning with a Learned Model. *Nature*, 588(7839): 604–609.
- Serrano, S.; and Smith, N. A. 2019. Is Attention Interpretable? In *Annual Meeting of the Association for Computational Linguistics (ACL)*.
- Shukor, M.; Aubakirova, D.; Capuano, F.; Kooijmans, P.; Palma, S.; Zouitine, A.; Aractingi, M.; Pascal, C.; Russi, M.; Marafioti, A.; et al. 2025. SmolVLA: A Vision-Language-Action Model for Affordable and Efficient Robotics. *arXiv preprint arXiv:2506.01844*.
- Team Wan; Wang, A.; Ai, B.; Wen, B.; Mao, C.; et al. 2025. Wan: Open and Advanced Large-Scale Video Generative Models. *arXiv preprint arXiv:2503.20314*.
- Vapnik, V.; and Vashist, A. 2009. A New Learning Paradigm: Learning Using Privileged Information. *Neural Networks*, 22(5).
- Wen, J.; Zhu, Y.; Li, J.; Tang, Z.; Shen, C.; and Feng, F. 2025. DexVLA: Vision-Language Model with Plug-In Diffusion Expert for General Robot Control. *arXiv preprint arXiv:2502.05855*.
- Won, J.; Lee, K.; Jang, H.; Kim, D.; and Shin, J. 2025. Dual-Stream Diffusion for World-Model Augmented Vision-Language-Action Model. *arXiv preprint arXiv:2510.27607*.
- Wu, H.; Jing, Y.; Cheang, C.; Chen, G.; Xu, J.; Li, X.; Liu, M.; Li, H.; and Kong, T. 2024. Unleashing Large-Scale Video Generative Pre-Training for Visual Robot Manipulation. In *International Conference on Learning Representations (ICLR)*.
- Ye, S.; Ge, Y.; Zheng, K.; Gao, S.; Yu, S.; Kurian, G.; et al. 2026. World Action Models Are Zero-Shot Policies. *arXiv preprint arXiv:2602.15922*.
- Yuan, T.; Dong, Z.; Liu, Y.; and Zhao, H. 2026. Fast-WAM: Do World Action Models Need Test-Time Future Imagination? *arXiv preprint arXiv:2603.16666*.
- Zhang, C.; Lu, R.; Tong, J.; Li, X.; Wang, Y.; and Li, H. 2026. EvoScene-VLA: Evolving Scene Beliefs Inside the Action Decoder for Chunked Robot Control. *arXiv preprint arXiv:2605.21862*.
- Zhang, W.; Liu, H.; Qi, Z.; Wang, Y.; Yu, X.; Zhang, J.; Dong, R.; He, J.; Wang, H.; Zhang, Z.; Yi, L.; Zeng, W.; and Jin, X. 2025. DreamVLA: A Vision-Language-Action Model Dreamed with Comprehensive World Knowledge. *CoRR*, abs/2507.04447.
- Zhao, Q.; Lu, Y.; Kim, M. J.; Fu, Z.; Zhang, Z.; Wu, Y.; Li, Z.; Ma, Q.; Han, S.; Finn, C.; Handa, A.; Liu, M.-Y.; Xiang, D.; Wetzstein, G.; and Lin, T.-Y. 2025. CoT-VLA: Visual Chain-of-Thought Reasoning for Vision-Language-Action Models. *arXiv preprint arXiv:2503.22020*.
- Zheng, R.; Wang, J.; Reed, S.; Bjorck, J.; Fang, Y.; Hu, F.; Jang, J.; Kundalia, K.; Lin, Z.; Magne, L.; et al. 2025. FLARE: Robot Learning with Implicit World Modeling. *arXiv preprint arXiv:2505.15659*.
- Zhou, S.; Du, Y.; Chen, J.; Li, Y.; Yeung, D.-Y.; and Gan, C. 2024. RoboDreamer: Learning Compositional World Models for Robot Imagination. *arXiv preprint arXiv:2404.12377*.
- Zhu, C.; Yu, R.; Feng, S.; Burchfiel, B.; Shah, P.; and Gupta, A. 2025. Unified World Models: Coupling Video and Action Diffusion for Pretraining on Large Robotic Datasets. *arXiv preprint arXiv:2504.02792*.
- Zitkovich, B.; Yu, T.; Xu, S.; Xu, P.; Xiao, T.; Xia, F.; Wu, J.; Wohlfart, P.; Welker, S.; Wahid, A.; et al. 2023. RT-2: Vision-Language-Action Models Transfer Web Knowledge to Robotic Control. In *Conference on Robot Learning (CoRL)*.

A Training implementation details

Loss normalization. For $\mathcal{L}_{\text{vid}}$, we average unreduced squared error over channels and space, then valid latent frames; for $\mathcal{L}_{\text{FM}}$, over each patch vector, then valid future patches. Both receive the native video-flow timestep weight per sample before the batch mean. For $\mathcal{L}_{\text{act}}$, we average over action dimensions, mask padding, average over the horizon, and apply the base action-timestep weight before the batch mean. The coefficients of $\mathcal{L}_{\text{vid}}$ and $\mathcal{L}_{\text{act}}$ are 1. λ_{FM} and λ_{probe} stay at 1 for the first 70% of the curriculum, then cosine-decay to 0.2; the stopped-gradient probe cannot affect the deployed representation.

Deployment-matched perturbation. After 70% of the curriculum, perturbation probability and standard deviation rise linearly from 0 to 0.3 and from 0 to 0.06 times the latent standard deviation, respectively. Perturbed rows retain $\mathcal{L}_{\text{FM}}$ and $\mathcal{L}_{\text{probe}}$ but are excluded from $\mathcal{L}_{\text{act}}$.

Diagnostic probe. An RMS-normalized linear probe gives $\hat{Y}_{\text{L2}} = g_{\omega}(\text{RMS}(\text{stopgrad}(S_{\phi})))$ with $\mathcal{L}_{\text{probe}} = \text{mean}((\hat{Y}_{\text{L2}} - Y)^2)$ over all batch prediction elements; gradients reach only g_{ω} .

Gradient routes. $\mathcal{L}_{\text{vid}}$ updates the video expert and output head, but not the action expert; $\mathcal{L}_{\text{act}}$ updates the action expert and, through attended video states, the video expert and anticipation tokens; $\mathcal{L}_{\text{FM}}$ updates the video expert, anticipation tokens, and FM head; detached $\mathcal{L}_{\text{probe}}$ updates only the linear probe.

Architecture and training configuration. All in-house models share Wan2.2-5B’s pretrained video DiT, text encoder, and video VAE (Team Wan et al. 2025). The action expert reuses the video branch at $d_a = 1024$ (1B action; 6B total), with horizon $H = 32$. Multi-camera images are concatenated before the VAE; video is temporally downsampled $4\times$ to 9 frames per chunk. Both branches use Fast-WAM’s continuous flow matching. We sample $u \sim \mathcal{U}[0, 1]$, set $\sigma = 5u/(1 + 4u)$ and $t = 1000\sigma$, and reuse the video scheduler for the FM head. Deployment uses 10 action flow-matching steps, classifier-free guidance 1.0, and no video denoising or VAE decoding. Training uses AdamW at 10^{-4} learning rate, 0.01 weight decay, cosine annealing, mixed precision, and gradient clipping at 1.0, for 20k LIBERO or 30k RoboTwin 2.0 steps. Relative to Fast-WAM (Yuan et al. 2026), RIFT preserves these settings and adds only anticipation tokens, $\mathcal{L}_{\text{FM}}$, and the diagnostic probe.

B Uncertainty for the intervention study

Figure 2 reports point estimates; table 3 gives EE-ADE intervals and the task-level resampling protocol.

C Per-suite LIBERO success rates

Table 4 reports the per-suite results used in the main-text ablation discussion.

Table 3: Uncertainty for the EE-ADE estimates in fig. 2. The table reports percentile task-cluster bootstrap 95% confidence intervals in centimeters. Each replicate resamples whole tasks with replacement and recomputes the task-macro mean, keeping all trials from a sampled task together. Original is omitted because its EE-ADE is zero by definition. Final-clean replay substitutes both final-step keys and values at every action-denoising step, as in fig. 2. All reported intervention estimates use 2,000 paired trials across all 40 LIBERO tasks. N/A entries are structural rather than missing runs: IDM-style models already consume one fixed final-clean future K/V cache, so final-clean replay coincides with their Original, and Cosmos-2 exposes only one future timestep and therefore admits no temporal swap.

Model	Intervention	EE-ADE	95% CI
Fast-WAM-Joint	Final clean K/V	1.908	[1.579, 2.260]
	Spatial shuffle	14.306	[12.649, 16.057]
	Temporal swap	15.606	[13.973, 17.189]
	Noise	17.221	[16.017, 18.405]
	Frozen present	18.255	[16.876, 19.655]
	Masked future	18.725	[17.499, 19.953]
Fast-WAM-IDM	Final clean K/V	N/A	N/A
	Spatial shuffle	15.293	[13.214, 17.418]
	Temporal swap	16.325	[15.068, 17.520]
	Noise	17.939	[17.070, 18.779]
	Frozen present	21.028	[19.901, 22.104]
	Masked future	20.378	[19.389, 21.370]
Cosmos-2	Final clean K/V	1.690	[0.260, 3.620]
	Spatial shuffle	4.976	[4.342, 5.643]
	Temporal swap	N/A	N/A
	Noise	10.494	[8.984, 12.034]
	Frozen present	18.090	[16.800, 19.323]
	Masked future	11.788	[10.516, 13.034]
LingBot-VA	Final clean K/V	N/A	N/A
	Spatial shuffle	19.820	[16.171, 22.550]
	Temporal swap	15.706	[12.171, 20.044]
	Noise	17.466	[16.099, 18.831]
	Frozen present	21.323	[17.515, 25.022]
	Masked future	19.176	[17.066, 21.131]

D LIBERO-Plus OOD robustness

We directly evaluate LIBERO-trained checkpoints on the separate LIBERO-Plus benchmark without further training. RIFT achieves 81.1% overall success, outperforming Fast-WAM-IDM by 9.7 percentage points. It leads the four evaluated checkpoints across all seven perturbation categories, all five labeled difficulty levels, and all four source suites; table 5 reports the full breakdown.

E Optional L2–FM uncertainty warning

The final recipe retains direct L2’s deterministic future-latent readout as a stopped-gradient diagnostic probe. Both heads read the same anticipation states without feeding the controller; this controller-independent comparison runs only as an optional monitor and is excluded from policy-only deployment and reported latency. Let $\mu_{\text{L2}} = \hat{Y}_{\text{L2}} \in \mathbb{R}^{m \times d_y}$ denote

Table 4: **LIBERO per-suite SR (%)**. Mean $\pm$ std over three seeds (500 trials/suite/seed; 2,000 overall). **Bold**: column bests.

Method	Spatial	Object	Goal	Long	Overall
Fast-WAM	97.0 $\pm$ 0.43	99.7 $\pm$ 0.09	96.7 $\pm$ 0.50	93.7 $\pm$ 0.57	96.8 $\pm$ 0.27
PFD	98.3 $\pm$ 0.19	99.2 $\pm$ 0.33	98.1 $\pm$ 0.62	93.7 $\pm$ 0.41	97.3 $\pm$ 0.12
Fast-WAM-Joint	99.3 $\pm$ 0.41	99.5 $\pm$ 0.25	98.1 $\pm$ 0.62	96.6 $\pm$ 0.57	98.4 $\pm$ 0.26
Fast-WAM-IDM	99.3 $\pm$ 0.23	99.3 $\pm$ 0.31	98.1 $\pm$ 1.03	97.5 $\pm$ 0.99	98.6 $\pm$ 0.34
Cosmos Policy	97.8 $\pm$ 0.43	100.0 $\pm$ 0.00	97.8 $\pm$ 0.16	97.4 $\pm$ 0.00	98.3 $\pm$ 0.11
LingBot-VA	98.5 $\pm$ 0.14	99.6 $\pm$ 0.18	97.2 $\pm$ 0.38	98.5 $\pm$ 0.17	98.5 $\pm$ 0.08
RIFT-L2	98.6 $\pm$ 0.43	99.7 $\pm$ 0.09	98.1 $\pm$ 0.34	97.2 $\pm$ 0.50	98.4 $\pm$ 0.12
RIFT (full)	98.7 $\pm$ 0.25	99.7 $\pm$ 0.09	98.8 $\pm$ 0.43	98.0 $\pm$ 0.28	98.8 $\pm$ 0.17

Table 5: **Out-of-distribution robustness on LIBERO-Plus**. Success rates (%) use the pinned 10,030-variant protocol with one rollout per variant (environment/policy seed 42; initial-state index 0) and the benchmark-provided task-language field. Panels report the official perturbation, difficulty, and source-suite breakdowns; parenthesized header values are variant counts. Overall is the variant-micro average rather than an unweighted average of groups. “Uncl.” contains the 121 variants whose official difficulty label is null. Every populated row is complete over all 10,030 variants.

(a) Perturbation category								
Method	Camera (1,599)	Robot (1,550)	Language (1,537)	Light (1,142)	Texture (1,076)	Noise (1,601)	Layout (1,525)	Overall (10,030)
Fast-WAM	16.51	43.42	67.79	80.12	51.58	37.85	61.18	49.73
Fast-WAM-Joint	38.71	63.87	93.10	94.66	55.67	57.71	77.51	68.06
Fast-WAM-IDM	46.22	69.10	94.21	92.82	56.41	61.77	81.51	71.36
RiFT (full)	59.97	77.48	97.20	96.58	75.09	81.64	82.56	81.07
(b) Difficulty								
Method	Level 1 (1,644)	Level 2 (2,202)	Level 3 (2,094)	Level 4 (1,886)	Level 5 (2,083)	Uncl. (121)		
Fast-WAM	68.92	57.90	50.67	44.17	28.61	74.38		
Fast-WAM-Joint	87.59	80.02	71.59	60.50	41.43	100.00		
Fast-WAM-IDM	89.54	83.47	73.97	64.21	46.42	99.17		
RiFT (full)	94.59	90.87	85.39	77.36	57.99	99.17		
(c) Source LIBERO suite								
Method	Spatial (2,402)	Object (2,518)	Goal (2,591)	Long (2,519)				
Fast-WAM	52.54	69.54	37.21	40.13				
Fast-WAM-Joint	81.39	72.08	56.43	63.28				
Fast-WAM-IDM	81.72	72.40	63.80	68.20				
RiFT (full)	88.72	87.81	73.52	74.79				

the probe estimate, let $x_i^{\text{FM}} \in \mathbb{R}^{m \times d_y}$ be the i th of K samples from the conditional-FM head, and let $\bar{x}_{\text{FM}} = K^{-1} \sum_i x_i^{\text{FM}}$ denote their sample mean at the current environment step. We measure their normalized cross-estimator discrepancy as

$$d_{\text{ratio}} = \frac{\|\mu_{\text{L2}} - \bar{x}_{\text{FM}}\|_F^2}{K^{-1} \sum_{i=1}^K \|x_i^{\text{FM}} - \bar{x}_{\text{FM}}\|_F^2 + \epsilon}, \quad (7)$$

where $\epsilon > 0$ stabilizes the denominator. Large d_{ratio} means that the probe point estimate departs from the FM mean beyond the dispersion of the FM samples. It is a warning statistic, not a failure probability: both heads may still agree on the same wrong future. Unlike the action-flow acceleration of Rao et al. (2026), this score compares two estimators in future-latent space. In fig. 7, a one-sided CUSUM accumulates

$$S_t = \max(0, S_{t-1} + d_{\text{ratio}}(t) - \mu_d - 0.25\sigma_d), \quad (8)$$

Here μ_d and σ_d are calibrated from the 1,967 successes in the 2,000-episode ledger, and η is the conformal CUSUM threshold. Figure 7 averages S_t over all 33 failed rollouts, which are excluded from calibration; the mean crosses η 210 steps before the common $t = 420$ endpoint, although individual crossing times can differ.

F Limitations and future work

Evaluation is simulation-only. Physical robots and non-WAM fusion backbones remain future work.

G RoboTwin results and future decodes

Table 6 reports per-task RoboTwin 2.0 results; figs. 8 and 9 compare matched Fast-WAM-Joint and RIFT decodes. Visualizations are diagnostic; section 3.5 gives closed-loop evidence.

Table 6: **Per-task RoboTwin 2.0 success rates (%) under clean and randomized settings.** * marks results from Yuan et al. (2026); bold marks each row’s best per setting.

Task	$\pi_{0.5}^*$		Motus*		LingBot-VA		Fast-WAM-Joint		Fast-WAM		PFD		RIFT (Ours)	
	Clean	Rand.	Clean	Rand.	Clean	Rand.	Clean	Rand.	Clean	Rand.	Clean	Rand.	Clean	Rand.
Adjust Bottle	100	99	89	93	90	94	99	98	100	100	100	100	100	100
Beat Block Hammer	96	93	95	88	95	98	100	99	100	97	100	97	100	97
Blocks Ranking RGB	92	85	99	97	98	98	100	100	100	98	100	98	100	98
Blocks Ranking Size	49	26	75	63	94	96	87	89	93	98	94	98	94	98
Click Alarmclock	98	89	100	100	98	99	99	100	100	100	100	100	100	100
Click Bell	99	66	100	100	99	99	100	97	100	100	100	100	100	100
Dump Bin Bigbin	92	97	95	91	89	96	96	94	97	96	97	96	97	97
Grab Roller	100	100	100	100	99	99	100	100	100	100	100	100	100	100
Handover Block	66	57	86	73	98	78	92	93	97	81	97	82	97	83
Handover Mic	98	97	78	63	94	96	100	100	99	100	99	100	99	100
Hanging Mug	18	17	38	38	40	28	63	68	60	62	63	65	65	67
Lift Pot	96	85	96	99	99	98	100	100	100	100	100	100	100	100
Move Can Pot	51	55	34	74	94	97	98	98	90	91	91	92	91	92
Move Pillbottle Pad	84	61	93	96	98	99	100	99	100	99	100	99	100	99
Move Playingcard Away	96	84	100	96	99	99	100	100	100	100	100	100	100	100
Move Stapler Pad	56	42	83	85	91	79	83	84	72	70	74	72	75	74
Open Laptop	90	96	95	91	92	94	91	90	99	100	99	100	99	100
Open Microwave	34	77	95	91	82	86	6	25	59	37	62	41	64	45
Pick Diverse Bottles	81	71	90	91	89	82	89	85	81	88	82	89	83	89
Pick Dual Bottles	93	63	96	90	99	99	97	100	100	96	100	96	100	97
Place A2B Left	87	82	88	79	96	93	97	95	94	95	94	95	95	96
Place A2B Right	87	84	91	87	96	95	94	96	94	96	94	96	95	97
Place Bread Basket	77	64	91	94	96	95	91	92	93	93	94	94	94	94
Place Bread Skillet	85	66	86	83	95	90	88	95	95	95	95	95	96	96
Place Burger Fries	94	87	98	98	96	95	100	99	94	98	94	98	95	98
Place Can Basket	62	62	81	76	81	84	53	35	68	63	70	65	72	68
Place Cans Plasticbox	94	84	98	94	99	99	99	97	100	97	100	97	100	97
Place Container Plate	99	95	98	99	98	97	98	100	97	100	97	100	97	100
Place Dual Shoes	75	75	93	87	94	89	95	87	94	89	94	90	95	90
Place Empty Cup	100	99	99	98	99	99	100	100	100	100	100	100	100	100
Place Fan	87	85	91	87	98	93	98	97	95	95	95	95	96	96
Place Mouse Pad	60	39	66	68	93	96	94	92	88	89	89	90	89	90
Place Object Basket	80	76	81	87	91	88	88	80	86	84	87	85	88	86
Place Object Scale	86	80	88	85	96	95	95	100	93	93	94	94	94	94
Place Object Stand	91	85	98	97	98	96	94	96	89	93	90	94	90	94
Place Phone Stand	81	81	87	86	96	97	100	100	99	99	99	99	99	99
Place Shoe	92	93	99	97	97	98	94	98	96	97	96	97	96	97
Press Stapler	87	83	93	98	85	82	52	58	94	96	94	96	95	97
Put Bottles Dustbin	84	79	81	79	87	91	95	93	91	88	92	89	92	89
Put Object Cabinet	80	79	88	71	85	87	93	91	93	90	94	91	94	91
Rotate QRcode	89	87	89	73	96	91	90	94	91	90	92	91	92	91
Scan Object	72	65	67	66	96	91	93	91	90	91	91	92	91	92
Shake Bottle	99	97	100	97	99	97	100	100	100	100	100	100	100	100
Shake Bottle Horizontally	99	99	100	98	99	99	99	100	100	100	100	100	100	100
Stack Blocks Three	91	76	91	95	98	98	99	96	95	96	95	96	96	97
Stack Blocks Two	97	100	100	98	99	98	100	100	100	100	100	100	100	100
Stack Bowls Three	77	71	79	87	86	83	87	84	78	81	80	82	81	83
Stack Bowls Two	95	96	98	98	94	98	96	97	93	94	94	94	94	95
Stamp Seal	79	55	93	92	96	97	97	98	89	97	90	97	90	97
Turn Switch	62	54	84	78	44	45	71	75	60	66	63	68	65	70
Average	82.7	76.8	88.7	87.0	92.4	91.4	91.0	91.1	91.9	91.6	92.5	92.1	92.9	92.6



Figure 8: **Imagined future: Fast-WAM-Joint iterative diffusion versus RIFT one pass.** Per state, the top strip is Fast-WAM-Joint’s future decoded from its full iterative diffusion rollout; the bottom strip is decoded from RIFT’s anticipation tokens after one backbone pass. The first column is the shared current observation at frame 0; the remaining columns show decoded frames 2, 4, 6, and 8. Visual similarity is illustrative; policy behavior is evaluated separately.

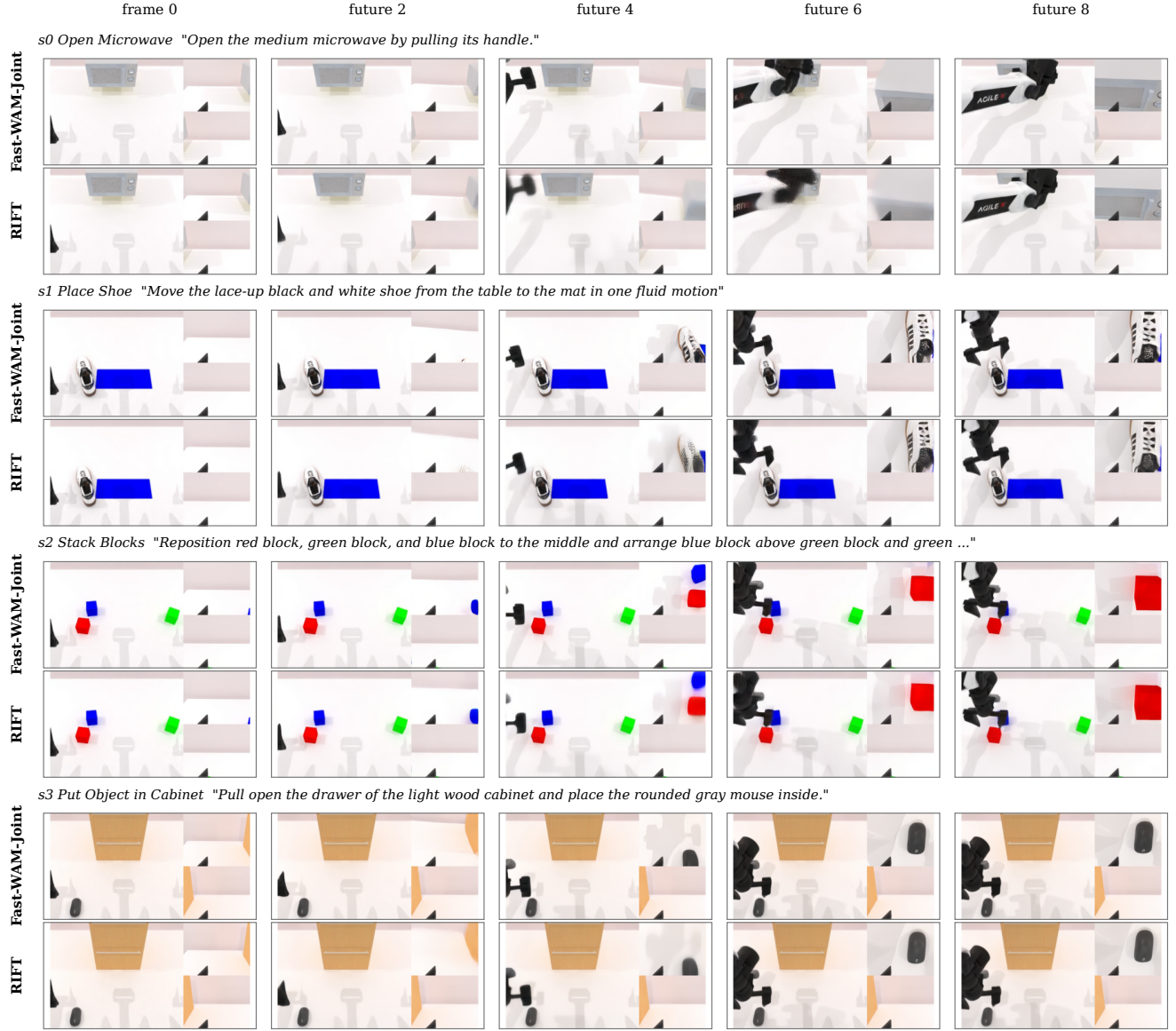


Figure 9: **RoboTwin 2.0 imagined future: Fast-WAM-Joint iterative diffusion versus RIFT one pass.** Each row pair starts from the same raw observation from three cameras. The first column is the shared observation at frame 0; the remaining columns show decoded frames 2, 4, 6, and 8. The top strip uses Fast-WAM-Joint’s iterative video diffusion with 20 denoising steps. The bottom strip shows RIFT’s anticipation states from one backbone pass, decoded by the trained linear diagnostic probe and frozen VAE. The RIFT video decode is diagnostic and is not part of its deployed action path; policy behavior is evaluated separately.