LEARNING FROM WHAT GOES WRONG: FAILURE-AWARE WORLD MODELS FOR FAITHFUL ROBOT POLICY EVALUATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Video world models offer a scalable proxy for robot policy evaluation, but success-only training can overestimate policy success when the supplied actions lead to failure. Simply adding failure examples is still insufficient, because the failure space is broad and post-failure object dynamics are difficult to model plausibly, often leading to unrealistic motion or appearance changes that can mislead downstream VLM-based judges. We present a failure-aware framework that combines simulator-verified, scene-matched success–failure trajectories with Semantic-Aware Feedback Forcing (SAFF). SAFF uses feedback and semantic-motion signals to prioritize interaction-relevant prediction errors without changing the flow-matching target or adding auxiliary inference models. We also introduce a six-dimensional benchmark with confidence-aware trajectory evaluation. On simulation and real-world video benchmarks, our model scores 73.17 and 73.31 out of 100, surpassing the strongest baseline by 5.32 and 21.21 points, respectively. Across five policies in each setting, world-model–VLM success-rate estimates achieve Pearson correlations of 0.928 and 0.942 with execution-based rates, while reducing mean absolute error relative to Ctrl-World from 13.0 to 7.0 and from 16.16 to 8.26 percentage points, respectively. These results demonstrate improved prediction fidelity and closer agreement with execution-based policy performance.

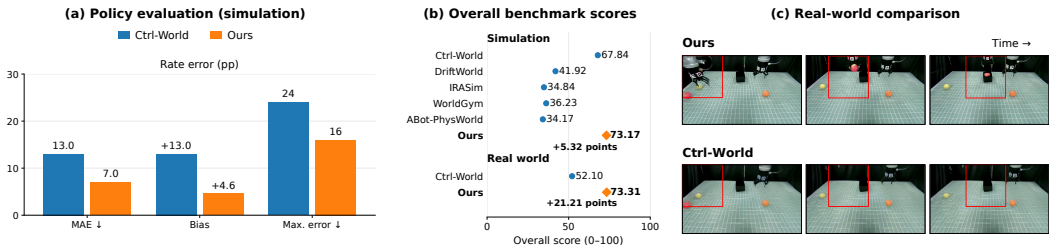

Figure 1: Failure-aware world modeling. (a) Simulation policy-evaluation errors in percentage points; bias closer to zero is better. (b) Overall benchmark scores. (c) Real-world fruit-to-box predictions: ours above Ctrl-World.

1 INTRODUCTION

Video generation has increasingly become a foundation for robotic world modeling. Early approaches such as Video Diffusion Models (Ho et al., 2022) advanced temporally coherent generation, while UniSim (Yang et al., 2024) and RoboDreamer (Zhou et al., 2024) extended generative prediction to interactive simulation and robot imagination. More recently, WorldEval (Li et al., 2025) and Ctrl-World (Guo et al., 2026) adapted video generators for action-conditioned policy evaluation, making generated rollouts a promising alternative to repeated physical deployment.

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

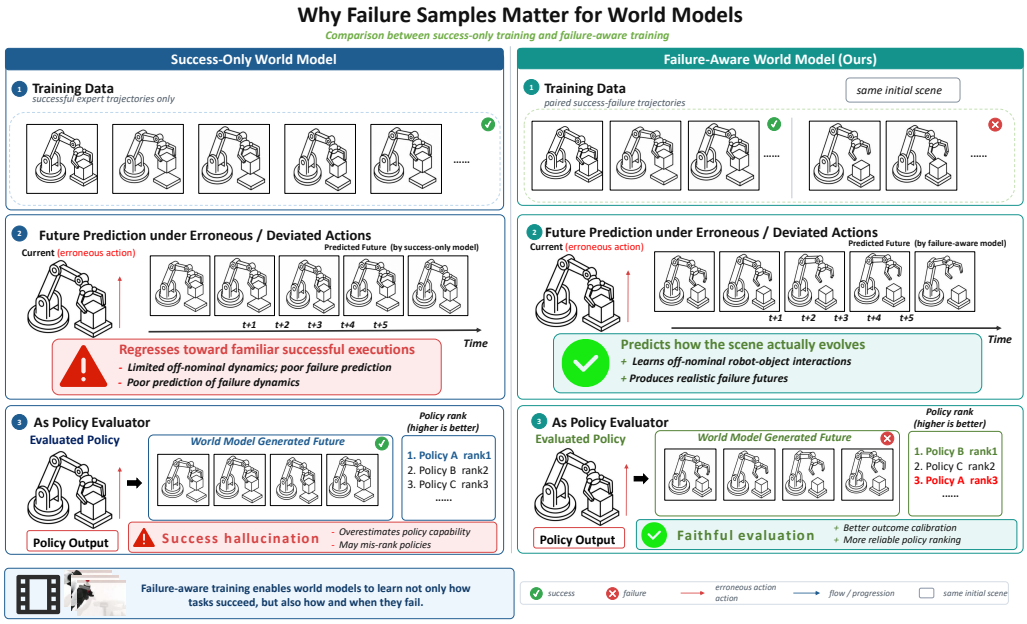

Figure 2: **Why failure examples matter for world models.** A conceptual comparison of success-only and failure-aware training across data coverage, action-conditioned future prediction, and downstream policy evaluation. Learning from failed executions aims to preserve the consequences of erroneous actions rather than hallucinate successful interactions.

However, visually plausible predictions need not faithfully reflect action consequences. As noted by Tseng et al. (2025), robot demonstration datasets are often dominated by successful executions, leaving limited coverage of erroneous actions and their outcomes. When a policy misses a grasp or releases an object prematurely, a success-biased model may nevertheless generate a successful interaction, inflating estimated success rates and undermining policy comparisons. An evaluator must predict what the supplied actions would cause, not merely how the task should be completed. This motivates two complementary challenges: *failure coverage*, which requires structured examples of how execution deviates from success, and *failure-dynamics learning*, which requires capturing the resulting changes in contact and object motion. Although non-expert rollout augmentation by Tseng et al. (2025) and failure-enriched training in dWorldEval (Li et al., 2026c) already address data coverage, our focus is to couple mechanism-specific failure construction with training signals that prioritize learning the resulting interaction dynamics.

We introduce a failure-aware world model (FAWM) that learns from both successful and failed trajectories. On RoboTwin 2.0 (Chen et al., 2025), we inject task-relevant perturbations into semantic motion primitives and verify failures in simulation, producing 133,500 failed rollouts across 50 tasks and 13 failure modes, paired with successful executions from the same initial scenes. Whereas REFLECT (Liu et al., 2023) selects event-based keyframes for failure explanation, we annotate a *perturbation-execution keyframe* marking when the deliberately introduced action perturbation is executed. This execution-grounded timestamp links the semantics of the perturbed action to its subsequent consequences. Figure 2 illustrates why this matters: changing the action should change the predicted interaction rather than reproduce a familiar successful future. Our model builds on Wan2.2 (Wan Team, 2025) and adopts Structured Kinematic-to-Visual Action Fields (KVAFs) from EA-WM (Yang et al., 2026) to express robot actions as camera-aligned visual conditions. To better learn failure dynamics, we introduce *Semantic-Aware Feedback Forcing* (SAFF). Inspired by feedback-based correction in FBFM (Li et al., 2026b), SAFF combines feedback-derived correction demand with semantic and motion sensitivity to identify informative spatiotemporal regions. These signals reweight the standard flow-matching loss, emphasizing interaction-relevant prediction errors without replacing its target or requiring auxiliary feedback and semantic models at inference.

We further introduce a six-dimensional benchmark with confidence-aware trajectory evaluation using SAM 3 (Carion et al., 2025). It distinguishes insufficient reference evidence from prediction-side tracking failures and complements execution-based policy evaluation with measures of visual and interaction fidelity.

- **Controlled failure data.** We develop a simulator-verified pipeline that constructs interpretable, same-scene success–failure pairs with perturbation-execution keyframe annotations, linking controlled action deviations to their subsequent consequences.
- **Failure-aware world modeling.** We introduce SAFF, which integrates feedback-derived correction demand and semantic-motion sensitivity into training-time loss reweighting for a KVAF-conditioned video world model.
- **Confidence-aware evaluation.** We develop a six-dimensional benchmark that combines trajectory confidence and explicit missing-data rules with complementary visual and motion metrics to assess prediction fidelity relevant to policy evaluation.

2 Related Work

Video-generative world models. Building on Video Diffusion Models (Ho et al., 2022), UniSim (Yang et al., 2024) and RoboDreamer (Zhou et al., 2024) advanced interactive simulation and compositional planning using mixed-domain data and RT-1 demonstrations, respectively. IRASim (Zhu et al., 2025) and WorldEval (Li et al., 2025) strengthened action conditioning through frame-level normalization and policy-derived representations, respectively, using robot interaction data. EA-WM (Yang et al., 2026) introduced camera-aligned KVAFs on RoboTwin. Complementarily, VideoREPA (Zhang et al., 2025) and Geometry Forcing (Wu et al., 2026) pursued physical consistency through relational and geometric alignment on OpenVid and RealEstate10K/Minecraft, respectively. These advances address controllability and coherence, but architectural improvements alone do not establish coverage of execution failures.

Failure-inclusive approaches address this data limitation: Ctrl-World (Guo et al., 2026) combines multi-view prediction and memory with successful and failed DROID trajectories, while Tseng et al. (2025) augment RoboMimic demonstrations with non-expert policy rollouts but report residual interaction hallucinations. World-VLA-Loop (Liu et al., 2026) jointly learns videos and rewards from its SANS dataset of successful, pose-perturbed, and natural-failure trajectories; dWorldEval (Li et al., 2026c) uses masked discrete diffusion over visual, action, and progress tokens with simulated and real failures. Relatedly, Foresight (Zhang et al., 2026) uses outcome-labeled world-model latents for failure monitoring, whereas FBFM (Li et al., 2026b) corrects generation using execution-time observations. These findings motivate our joint study of mechanism-specific failure coverage and training signals that prioritize interaction-relevant errors through feedback and semantic-motion sensitivity.

Constructing robotic failure trajectories. FINO-Net (Inceoglu et al., 2021) and REFLECT (Liu et al., 2023) established failure resources for multimodal detection and event-based explanation, respectively. AHA's FailGen (Duan et al., 2025) scales failure synthesis through keyframe-based perturbations of RLBench demonstrations, including parameter sweeps and simulator outcome checks; Guardian (Pacaud et al., 2026) combines online simulator perturbations with offline transformations of real-robot examples for failure reasoning. I-FailSense (Grislain et al., 2025) and ViFailback (Zeng et al., 2026) enrich supervision through semantic-misalignment examples and visually annotated real-robot failures, respectively. Dream2Fix (Li et al., 2026a) instead synthesizes failure–correction pairs inside a learned world model, avoiding simulator assets but depending on the generator's fidelity. These efforts primarily support detection, explanation, or recovery. Building on structured failure synthesis, our data construction emphasizes task applicability, manipulation-phase-specific physical interventions, and validation of both intervention execution and task outcome. Same-scene success–failure pairs with *perturbation-execution keyframes* connect identifiable action deviations to subsequent world evolution for action-conditioned prediction.

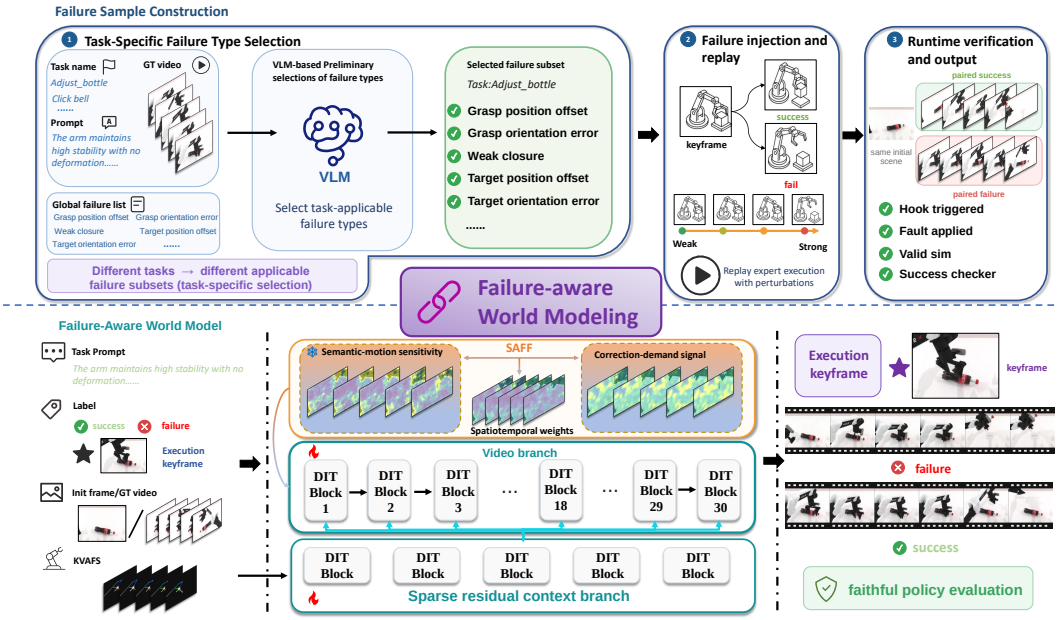

Figure 3: Failure-aware world modeling. Verified perturbations produce scene-matched success–failure pairs with execution-keyframe annotations. The KVAF-conditioned video model is trained using detached SAFF weights. Ground-truth futures and auxiliary feedback/semantic computations are used only during training; outcome and keyframe annotations are not inference conditions.

3 METHOD

Given an initial observation, a task prompt, and robot actions, our framework learns to predict successful and failed futures through controlled data construction and SAFF (Figure 3), evaluated with a confidence-aware benchmark.

3.1 CONTROLLED FAILURE TRAJECTORY CONSTRUCTION

Building on AHA/FailGen (Duan et al., 2025), we construct scene-matched success–failure pairs on RoboTwin 2.0 (Chen et al., 2025) (Table 1). A VLM selects task-applicable subsets of 13 physical failure modes from task descriptions and successful videos; runtime hooks inject them into phase-specific manipulation primitives during expert replay. We increase perturbation magnitude until the first valid failure in the tested sequence, verifying hook activation, fault application, simulation validity, and failure under the original task checker while rejecting pre-intervention failures and artifacts. Each pair preserves the initial scene and source-episode identity and includes a *perturbation-execution keyframe* marking when the modified action is executed. The pipeline yields 133,500 accepted failures from 25,000 successful episodes across 50 tasks; construction details and comparison criteria appear in Appendix A.1.

3.2 FAILURE-AWARE WORLD MODELING WITH SAFF

Our model combines Wan2.2 TI2V-5B (Wan Team, 2025) with camera-aligned KVAFs from EA-WM (Yang et al., 2026), injected through a sparse residual context branch. After training the KVAF branch for action conditioning, we fine-tune the video backbone's LoRA adapters (Hu et al., 2022) using SAFF. For a target video x with latent z_0, standard flow matching (Lipman et al., 2023) uses $z_\sigma = (1-\sigma)z_0 + \sigma\epsilon$ and $v^\star = \epsilon - z_0$, where $\epsilon \sim \mathcal{N}(0, I)$. Each successful or failed trajectory supplies its own target, and the conditioned initial-frame slot is excluded from future supervision.

Table 1: Comparison of failure-data construction designs. ✓: explicit support; △: partial or related support; ×: not explicitly addressed under the criteria in Appendix A.1. Symbols describe components, not overall method quality.

Method	Failure taxonomy	Phase-aware modes	Task–fault filtering	Primitive injection	Adaptive severity search	Sim. outcome check
REFLECT/RoboFail (Liu et al., 2023)	✓	×	×	×	×	×
AHA/FailGen (Duan et al., 2025)	✓	△	×	△	△	✓
Guardian/FailCoT (Pacaud et al., 2026)	✓	△	×	△	×	△
ViFailback (Zeng et al., 2026)	✓	△	×	×	×	×
Ours	✓	✓	✓	✓	✓	✓

The current prediction gives the endpoint $\hat{z}_0 = z_\sigma - \sigma v_\theta$. Following FBFM (Li et al., 2026b), we measure *correction demand* through an endpoint vector–Jacobian product:

$$\delta v = -\lambda_\sigma M J_\sigma^\top M(z_0 - \hat{z}_0), \qquad J_\sigma = \frac{\partial \hat{z}_0}{\partial z_\sigma}, \tag{1}$$

where M masks out the conditioned slot and λ_σ is the feedback gain. The residual is held fixed when computing the vector–Jacobian product. Channel-wise RMS, spatiotemporal smoothing, and 90th-percentile normalization with clipping over future positions yield a correction-demand map $C \in [0, 1]$.

To obtain complementary *semantic-motion sensitivity*, frozen DINOv3 (Siméoni et al., 2025) extracts unit-normalized features $\hat{f}^l_{s,j}$ and $f^l_{s,j}$ from decoded predictions and ground truth, indexed by sampled frame s, patch j, and layer l. Across L selected layers, we compute

$$\begin{aligned} a_{s,j} &= \frac{1}{L}\sum_l \left(1 - \langle \hat{f}^l_{s,j}, f^l_{s,j}\rangle\right), \\ b_{s,j} &= \frac{1}{L}\sum_l \mathrm{RMS}\Big[(\hat{f}^l_{s,j} - \hat{f}^l_{s-1,j}) - (f^l_{s,j} - f^l_{s-1,j})\Big]. \end{aligned} \tag{2}$$

These terms capture appearance discrepancies and mismatched temporal feature changes, respectively. With per-frame quantile scales u_s, v_s and detached candidate weights $c_{s,j}$ derived from their joint responses, the auxiliary objective over S sampled frames is

$$\mathcal{L}_{\mathrm{sm}} = \frac{1}{S-1}\sum_{s=1}^{S-1} \frac{\sum_j \mathrm{sg}(c_{s,j})\, \frac{1}{2}(a_{s,j}/\,\mathrm{sg}(u_s) + b_{s,j}/\,\mathrm{sg}(v_s))}{\max\Big(\sum_j \mathrm{sg}(c_{s,j}), 10^{-6}\Big)}. \tag{3}$$

Here sg stops gradients; Appendix A.2 specifies the scales and candidate construction. We project the resulting gradient through the frozen VAE decoder $\mathcal{D}$:

$$g_{\mathrm{sm}} = \left(\frac{\partial \mathcal{D}(\hat{z}_0)}{\partial \hat{z}_0}\right)^\top \nabla_{\hat{x}} \mathcal{L}_{\mathrm{sm}}(\hat{x}, x), \qquad \hat{x} = \mathcal{D}(\hat{z}_0). \tag{4}$$

Channel-wise RMS and spatial 90th-percentile normalization with clipping within each future latent time slot produce $D \in [0, 1]$.

For future latent positions $i \in \mathcal{U}$, the two maps determine detached weights and the training loss:

$$\begin{aligned} W_i &= \mathrm{sg}\big[1 + (w_{\max} - 1)C_i\big(\alpha + (1-\alpha)D_i\big)\big], \\ \mathcal{L}_{\mathrm{SAFF}} &= \mathbb{E}\left[\omega(\sigma)\frac{\sum_{i\in\mathcal{U}} W_i\, \ell_i^{\mathrm{FM}}}{\sum_{i\in\mathcal{U}} W_i}\right]. \end{aligned} \tag{5}$$



Figure 4: Real-world task setups. From left to right: placing fruit in a black box, placing a toy duck on a shelf, and hanging a mug on a mug tree.

Here ℓ_i^{FM} is the channel-mean squared velocity error and $\omega(\sigma)$ is the scheduler weight. We use $\alpha = 0.1$ and $w_{\max} = 2$, retaining correction evidence when semantic sensitivity is weak. Weight computation and training share the same parameters, noise, and conditions. Only the weighted FM loss updates the model; $\mathcal{L}_{\mathrm{sm}}$ constructs weights rather than adding a separate training loss. The velocity target is unchanged, and inference requires no feedback or DINO computations.

3.3 Confidence-Aware World-Model Benchmark

Under shared spatial and temporal alignment, we evaluate *visual fidelity* with PSNR, SSIM (Wang et al., 2004), and LPIPS (Zhang et al., 2018); *distributional quality* with dataset-level FVD (Unterthiner et al., 2018); *temporal consistency* with GT-baseline-corrected RAFT (Teed & Deng, 2020) flow warping; *semantic consistency* with DINOv2 (Oquab et al., 2023) over scene and motion-derived object regions; *motion consistency* with camera-compensated motion-mask ST-IoU; and *trajectory consistency* with SAM 3 (Carion et al., 2025) arm/object centroid tracks. The trajectory score combines confidence-weighted localization errors with penalties for low-confidence agreement and missing observations. Insufficient GT evidence causes the corresponding role to abstain, whereas missing or insufficient predicted tracks score zero when GT support is adequate. The six normalized dimensions form a 0–100 fidelity score with explicit missing-data handling; metric formulas, thresholds, and aggregation rules are provided in Appendix A.3.

4 Experiments

4.1 Experimental Setup

Datasets. *Simulation.* We use our RoboTwin 2.0 (Chen et al., 2025) corpus, comprising 25,000 successful expert trajectories and 133,500 simulator-verified failed rollouts across 50 tasks. The failures cover 13 physical modes and are paired with successful executions from the same initial scenes, as described in Section 3.1. *Real world.* We collect teleoperated trajectories using two Flexiv Rizon 4s arms, each with seven degrees of freedom and a Flexiv Grav (Flexiv-GN01) two-finger force-controlled gripper. The three tasks are placing fruit in a black box, placing a toy duck on a shelf, and hanging a mug by its handle on a mug tree (Figure 4). Videos and robot states are synchronized at 30 Hz, and camera-aligned KVAFs are generated from joint and gripper states.

Training protocol. We train our world model on NVIDIA H100 GPUs with a batch size of 32 for 300,000 optimization steps. For policy evaluation, we train five $\pi_{0.5}$ (Physical Intelligence et al., 2025) policies using 10%, 20%, 30%, 50%, and 100% of the policy-training data, respectively. Each policy is trained for 50,000 steps on eight NVIDIA B300 GPUs with a batch size of 8. This varies the policy-training data budget while keeping the number of optimization steps fixed.

4.2 Main Results

Quantitative analysis. We first evaluate video prediction quality on 315 held-out simulation trajectories, comprising 50 successful and 265 failed executions, with episode-disjoint training and test

splits. We compare against Ctrl-World, DriftWorld (Lu et al., 2026), IRASim, WorldGym (Quevedo et al., 2026), and ABot-PhysWorld (Chen et al., 2026). All baselines are fine-tuned on our training data following their official procedures, and all models are evaluated under the same protocol.

Our model achieves an overall score of 73.17 in simulation, exceeding Ctrl-World by 5.32 points and leading five of six normalized dimensions (Table 2). FVD decreases from 98.1 to 55.9 and LPIPS from 0.1141 to 0.0767, while motion ST-IoU remains slightly lower (0.6025 vs. 0.6119). On 206 real-world video pairs, our model improves all six dimensions and raises the overall score from 52.10 to 73.31 (Table 3). Tables 2 and 3 report score-contributing metrics; normalized dimension scores and diagnostic measurements appear in Appendix A.4.

Table 2: Video prediction on 315 simulation pairs. Scene/Obj. are DINO similarities; Arm/Obj. are confidence-aware trajectory scores. Total (0–100) aggregates six normalized dimensions. Best and second-best results are **bold** and underlined.

	Visual			Dist.	Temp.	Semantic		Motion	Trajectory		
Model	PSNR ↑	SSIM ↑	LPIPS ↓	FVD ↓	Flow Δ ↓	Scene ↑	Obj. ↑	ST-IoU ↑	Arm ↑	Obj. ↑	Total ↑
Ctrl-World	22.49	0.8603	0.1141	98.1	0.00473	0.7953	**0.7597**	**0.6119**	0.5937	0.6143	67.84
DriftWorld	15.98	0.7306	0.2704	256.5	0.01818	0.5435	0.4360	0.3883	0.1152	0.4493	41.92
IRASim	12.65	0.6744	0.3387	262.4	0.02775	0.5347	0.3816	0.2339	0.1328	0.4167	34.84
WorldGym	13.06	0.7305	0.3324	511.8	0.01075	0.5464	0.3674	0.0541	0.0858	0.4717	36.23
ABot-PhysWorld	13.86	0.7490	0.3223	433.5	0.02624	0.5762	0.3931	0.0915	0.1514	0.4239	34.17
Ours	**25.10**	**0.8918**	**0.0767**	**55.9**	**0.00320**	**0.8008**	0.7566	0.6025	**0.6059**	**0.6898**	**73.17**

Table 3: Video prediction on 206 real-world pairs. Metric definitions follow Table 2. Best results are **bold**.

	Visual			Dist.	Temp.	Semantic		Motion	Trajectory		
Model	PSNR ↑	SSIM ↑	LPIPS ↓	FVD ↓	Flow Δ ↓	Scene ↑	Obj. ↑	ST-IoU ↑	Arm ↑	Obj. ↑	Total ↑
Ctrl-World	20.98	0.6016	0.3221	594.7	0.00188	0.7241	0.6161	0.4040	0.5624	0.3110	52.10
Ours	**24.18**	**0.8659**	**0.0577**	**94.3**	**0.00022**	**0.8905**	**0.7257**	**0.6145**	**0.8129**	**0.6426**	**73.31**

We evaluate the five $\pi_{0.5}$ policies in each environment using GPT-5.6 (OpenAI, 2026), with 17 fixed frames per 81-frame rollout and a shared prompt. Tables 14 and 15 show overestimation under success-only training and underestimation under standard FM. In simulation (Figure 5(a,b)), SAFF and Ctrl-World achieve similar correlations, but SAFF reduces MAE from 13.0 to 7.0 percentage points and signed bias from $+13.0$ to $+4.6$ points (Table 4).

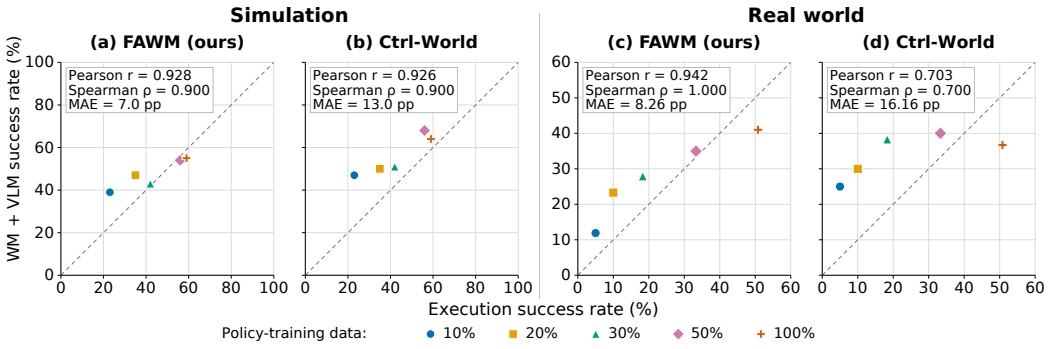

Figure 5: Policy evaluation in simulation (a,b) and the real world (c,d), comparing FAWM with Ctrl-World. Each marker represents one policy; axes compare execution-based and world-model–VLM success rates. Dashed lines indicate exact agreement. Axis ranges differ between environments.

Table 4: Policy-level agreement across five policies. MAE, signed bias, and maximum absolute error are in percentage points (pp). Lower errors and bias closer to zero are better.

Model	Pearson r	Spearman ρ	MAE (pp) ↓	Bias (pp; ideal 0)	Max. error (pp) ↓
Ctrl-World	0.926	0.90	13.0	+13.0	24
Ours	**0.928**	0.90	**7.0**	**+4.6**	**16**

Table 5: Simulation ablations using the metrics of Table 2. Normalized dimension scores are reported in Table 10.

	Visual			Dist.	Temp.	Semantic		Motion	Trajectory		
Variant	PSNR ↑	SSIM ↑	LPIPS ↓	FVD ↓	Flow Δ ↓	Scene ↑	Obj. ↑	ST-IoU ↑	Arm ↑	Obj. ↑	Total ↑
w/o both	15.37	0.7537	0.2612	199.1	0.02309	0.6413	0.4862	0.1908	0.2824	0.4608	40.78
w/o SAFF	22.99	0.8752	0.0943	71.1	0.00344	0.8004	0.7538	0.5913	0.5805	0.6780	70.89
Ours	25.10	0.8918	0.0767	55.9	0.00320	0.8008	0.7566	0.6025	0.6059	0.6898	73.17

In the real world (Figure 5(c,d)), Pearson correlation increases from 0.703 to 0.942 and Spearman correlation from 0.70 to 1.00, while MAE decreases from 16.16 to 8.26 percentage points. These results indicate improved policy ordering and more accurate success-rate estimates for the five evaluated policies. The visual-judging protocol and additional policy-evaluation statistics are provided in Appendix A.4.

Qualitative analysis. Figure 6 compares Ctrl-World with our model on a cube-manipulation sequence. Our prediction better preserves the cube's appearance and post-contact displacement relative to ground truth. The real-world fruit-to-box comparison in Figure 7 similarly shows closer agreement with the reference interaction, with detailed analysis and the full multi-model comparison in Appendix A.5.

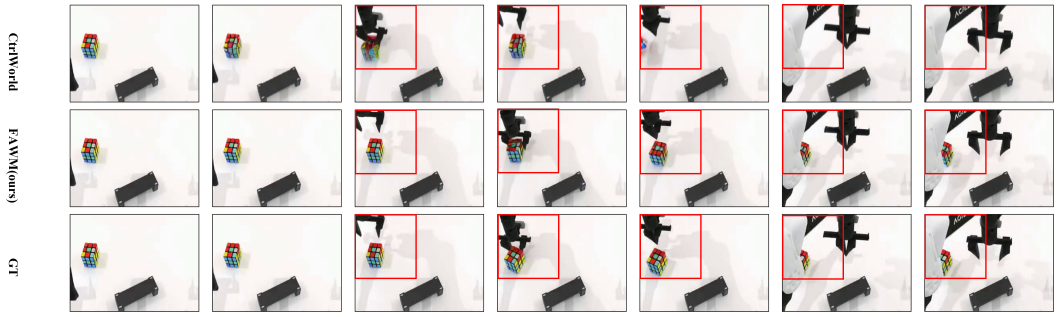

Figure 6: Qualitative comparison on a simulated cube-manipulation sequence. Rows: Ctrl-World, ours, and ground truth (GT). Time progresses from left to right; red boxes highlight the interaction region. The full multi-model comparison is provided in Figure 9.

5 Ablation Studies

We evaluate the contribution of SAFF on the simulation benchmark (Table 5). The *w/o SAFF* variant retains KVAF conditioning and uses standard flow matching; *w/o both* removes KVAF conditioning and SAFF as a joint-removal reference. Figure 8 shows spurious object-like structures near the gripper under standard FM, whereas SAFF better preserves the cube's appearance and post-contact displacement relative to GT.

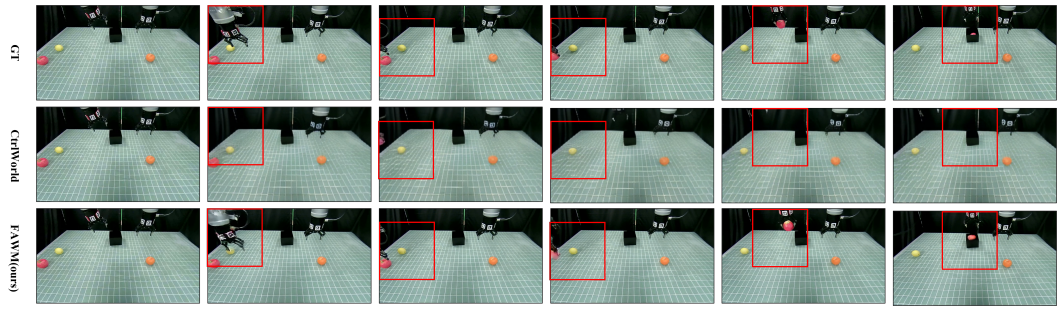

Figure 7: Real-world fruit-to-box predictions. Rows: GT, Ctrl-World, and ours. Time runs left to right; red boxes mark the interaction region.

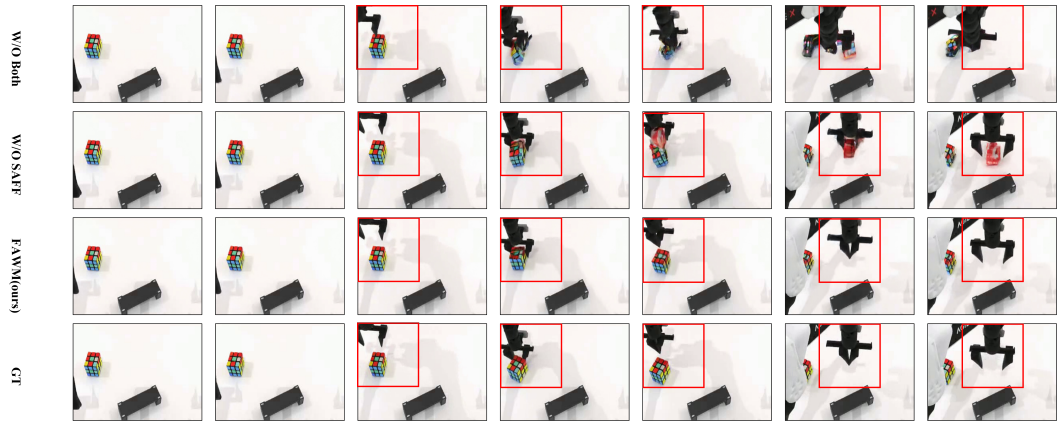

Figure 8: Qualitative ablations on a cube-manipulation sequence. Rows: w/o both, w/o SAFF, ours, and GT. Time progresses from left to right; red boxes highlight the interaction region.

6 Discussion and Limitations

Although our framework improves policy-evaluation reliability, discrepancies in absolute success-rate estimates remain, and strong correlation does not guarantee numerical agreement with actual execution. Moreover, the VLM judge may misinterpret task completion despite a standardized evaluation protocol. The reported results therefore reflect the combined world-model–VLM pipeline, highlighting the need to improve both prediction fidelity and judge reliability.

7 Conclusion

We presented a failure-aware world modeling framework that combines controlled success–failure data with SAFF. Our pipeline constructs simulator-verified trajectory pairs with perturbation-execution keyframes, while SAFF uses feedback and semantic-motion signals to emphasize interaction-relevant prediction errors during training. We further introduced a six-dimensional benchmark with confidence-aware trajectory evaluation. Experiments in simulation and real-world manipulation show improved video prediction fidelity and reduced policy success-rate estimation errors. These results highlight the value of structured failure coverage and targeted learning of failure dynamics for more faithful robot policy evaluation.

AI USE STATEMENT

Generative AI tools assisted with manuscript translation, language editing, and LaTeX preparation. VLMs also supported task–fault applicability selection and binary visual outcome assessment within the research pipeline, as described in Section 3.1 and Appendix A.4. The authors reviewed and revised the AI-assisted manuscript and take responsibility for the final content, including its claims, references, reported results, and research artifacts.

ETHICS STATEMENT

This work aims to improve the reliability of robot policy evaluation. Controlled failure injection is performed in simulation; world-model predictions and VLM judgments remain imperfect evaluation proxies rather than safety guarantees. Deployment decisions should therefore include physical validation and appropriate human oversight. Any distribution of data or models should respect the licenses of underlying resources and the privacy of individuals involved in data collection.

REPRODUCIBILITY STATEMENT

Section 4.1 reports the datasets and training protocol. Appendices A.1, A.2, and A.3 detail failure generation, SAFF implementation, and benchmark computation, including confidence penalties and missing-observation handling. Appendix A.4 provides additional metrics, policy-level results, and the fixed frame-sampling and VLM-judging protocol.

REFERENCES

Nicolas Carion, Laura Gustafson, Yuan-Ting Hu, Shoubhik Debnath, Ronghang Hu, Didac Suris, Chaitanya Ryali, Kalyan Vasudev Alwala, Haitham Khedr, Andrew Huang, Jie Lei, Tengyu Ma, Baishan Guo, Arpit Kalla, Markus Marks, Joseph Greer, Meng Wang, Peize Sun, Roman Rädle, Triantafyllos Afouras, Effrosyni Mavroudi, Katherine Xu, Tsung-Han Wu, Yu Zhou, Liliane Momeni, Rishi Hazra, Shuangrui Ding, Sagar Vaze, Francois Porcher, Feng Li, Siyuan Li, Aishwarya Kamath, Ho Kei Cheng, Piotr Dollár, Nikhila Ravi, Kate Saenko, Pengchuan Zhang, and Christoph Feichtenhofer. SAM 3: Segment anything with concepts. *arXiv preprint arXiv:2511.16719*, 2025. doi: 10.48550/arXiv.2511.16719. URL https://arxiv.org/abs/2511.16719.

Tianxing Chen, Zanxin Chen, Baijun Chen, Zijian Cai, Yibin Liu, Zixuan Li, Qiwei Liang, Xianliang Lin, Yiheng Ge, Zhenyu Gu, Weiliang Deng, Yubin Guo, Tian Nian, Xuanbing Xie, Qiangyu Chen, Kailun Su, Tianling Xu, Guodong Liu, Mengkang Hu, Huan-ang Gao, Kaixuan Wang, Zhixuan Liang, Yusen Qin, Xiaokang Yang, Ping Luo, and Yao Mu. RoboTwin 2.0: A scalable data generator and benchmark with strong domain randomization for robust bimanual robotic manipulation. *arXiv preprint arXiv:2506.18088*, 2025. doi: 10.48550/arXiv.2506.18088. URL https://arxiv.org/abs/2506.18088.

Yuzhi Chen, Ronghan Chen, Dongjie Huo, Yandan Yang, Dekang Qi, Haoyun Liu, Tong Lin, Shuang Zeng, Junjin Xiao, Xinyuan Chang, Feng Xiong, Xing Wei, Zhiheng Ma, and Mu Xu. ABot-PhysWorld: Interactive world foundation model for robotic manipulation with physics alignment, 2026. URL https://arxiv.org/abs/2603.23376.

Jiafei Duan, Wilbert Pumacay, Nishanth Kumar, Yi Ru Wang, Shulin Tian, Wentao Yuan, Ranjay Krishna, Dieter Fox, Ajay Mandlekar, and Yijie Guo. AHA: A vision-language-model for detecting and reasoning over failures in robotic manipulation. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/70a06501001e1820fd1eb9ee821302d2-Abstract-Conference.html.

Clémence Grislain, Hamed Rahimi, Olivier Sigaud, and Mohamed Chetouani. I-FailSense: Towards general robotic failure detection with vision-language models. *arXiv preprint arXiv:2509.16072*, 2025. doi: 10.48550/arXiv.2509.16072. URL https://arxiv.org/abs/2509.16072.

Yanjiang Guo, Lucy Xiaoyang Shi, Jianyu Chen, and Chelsea Finn. Ctrl-World: A controllable generative world model for robot manipulation. In *International Conference on Learning Representations*, 2026. URL https://arxiv.org/abs/2510.10125.

Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J. Fleet. Video diffusion models. In *Advances in Neural Information Processing Systems*, volume 35, 2022. doi: 10.52202/068431-0628. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/39235c56aef13fb05a6adc95eb9d8d66-Abstract-Conference.html.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Arda Inceoglu, Eren Erdal Aksoy, Abdullah Cihan Ak, and Sanem Sariel. FINO-Net: A deep multimodal sensor fusion framework for manipulation failure detection. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 6841–6847. IEEE, 2021. doi: 10.1109/IROS51168.2021.9636455. URL https://doi.org/10.1109/IROS51168.2021.9636455.

Dayou Li, Jiuzhou Lei, Hao Wang, Lulin Liu, Yunhao Yang, Zihan Wang, Bangya Liu, Minghui Zheng, and Zhiwen Fan. Learning actionable manipulation recovery via counterfactual failure synthesis. *arXiv preprint arXiv:2603.13528*, 2026a. doi: 10.48550/arXiv.2603.13528. URL https://arxiv.org/abs/2603.13528.

Peize Li, Ruimeng Zhang, Ru Zhang, Cong Huang, Kai Chen, and Shanghang Zhang. FBFM: A training-free asynchronous feedback mechanism for flow-matching in world-action models execution. *arXiv preprint arXiv:2607.29235*, 2026b. URL https://arxiv.org/abs/2607.29235.

Yaxuan Li, Yichen Zhu, Junjie Wen, Chaomin Shen, and Yi Xu. WorldEval: World model as real-world robot policies evaluator. *arXiv preprint arXiv:2505.19017*, 2025. doi: 10.48550/arXiv.2505.19017. URL https://arxiv.org/abs/2505.19017.

Yaxuan Li, Zhongyi Zhou, Yefei Chen, Yaokai Xue, and Yichen Zhu. dWorldEval: Scalable robotic policy evaluation via discrete diffusion world model. In *International Conference on Machine Learning*, 2026c. URL https://arxiv.org/abs/2604.22152.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2210.02747.

Xiaokang Liu, Zechen Bai, Hai Ci, Kevin Yuchen Ma, and Mike Zheng Shou. World-VLA-Loop: Closed-loop learning of video world model and VLA policy. *arXiv preprint arXiv:2602.06508*, 2026. doi: 10.48550/arXiv.2602.06508. URL https://arxiv.org/abs/2602.06508.

Zeyi Liu, Arpit Bahety, and Shuran Song. REFLECT: Summarizing robot experiences for failure explanation and correction. In *Proceedings of The 7th Conference on Robot Learning*, volume 229 of *Proceedings of Machine Learning Research*, pp. 3468–3484. PMLR, 2023. URL https://proceedings.mlr.press/v229/liu23g.html.

Susie Lu, Haonan Chen, Weirui Ye, and Yilun Du. DriftWorld: Fast world modeling through drifting, 2026. URL https://arxiv.org/abs/2607.15065.

OpenAI. GPT-5.6: Frontier intelligence that scales with your ambition. Model release announcement, July 2026. URL https://openai.com/index/gpt-5-6/.

Maxime Oquab, Timothée Darcet, Theo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Russell Howes, Po-Yao Huang, Hu Xu, Vasu Sharma, Shang-Wen Li, Wojciech Galuba, Mike Rabbat, Mido Assran, Nicolas Ballas, Gabriel Synnaeve, Ishan Misra, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision, 2023. URL https://arxiv.org/abs/2304.07193.

Paul Pacaud, Ricardo Garcia Pinel, Shizhe Chen, and Cordelia Schmid. Guardian: Detecting robotic manipulation failures with vision-language models. *arXiv preprint arXiv:2512.01946*, 2026. URL https://www.di.ens.fr/willow/research/guardian/.

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, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Allen Z. Ren, Lucy Xiaoyang Shi, Laura Smith, Jost Tobias Springenberg, Kyle Stachowicz, James Tanner, Quan Vuong, Homer Walke, Anna Walling, Haohuan Wang, Lili Yu, and Ury Zhilinsky. $\pi_{0.5}$: a vision-language-action model with open-world generalization, 2025. URL https://arxiv.org/abs/2504.16054.

Julian Quevedo, Ansh Kumar Sharma, Yixiang Sun, Varad Suryavanshi, Percy Liang, and Sherry Yang. WorldGym: World model as an environment for policy evaluation. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/7f5e909ac0324db03506b380c695ffaf-Abstract-Conference.html.

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

Zachary Teed and Jia Deng. RAFT: Recurrent all-pairs field transforms for optical flow. In *Computer Vision – ECCV 2020*, 2020. URL https://www.ecva.net/papers/eccv_2020/papers_ECCV/html/3526_ECCV_2020_paper.php.

Wei-Cheng Tseng, Jinwei Gu, Qinsheng Zhang, Hanzi Mao, Ming-Yu Liu, Florian Shkurti, and Yen-Chen Lin. Scalable policy evaluation with video world models. *arXiv preprint arXiv:2511.11520*, 2025. doi: 10.48550/arXiv.2511.11520. URL https://arxiv.org/abs/2511.11520.

Thomas Unterthiner, Sjoerd van Steenkiste, Karol Kurach, Raphael Marinier, Marcin Michalski, and Sylvain Gelly. Towards accurate generative models of video: A new metric & challenges. *arXiv preprint arXiv:1812.01717*, 2018. URL https://arxiv.org/abs/1812.01717.

Wan Team. Wan2.2, 2025. URL https://github.com/Wan-Video/Wan2.2. Official model and code release.

Zhou Wang, Alan C. Bovik, Hamid R. Sheikh, and Eero P. Simoncelli. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4): 600–612, 2004. doi: 10.1109/TIP.2003.819861. URL https://doi.org/10.1109/TIP.2003.819861.

Haoyu Wu, Diankun Wu, Tianyu He, Junliang Guo, Yang Ye, Yueqi Duan, and Jiang Bian. Geometry Forcing: Marrying video diffusion and 3D representation for consistent world modeling. In *International Conference on Learning Representations*, 2026. URL https://arxiv.org/abs/2507.07982.

Sherry Yang, Yilun Du, Seyed Ghasemipour, Jonathan Tompson, Leslie Kaelbling, Dale Schuurmans, and Pieter Abbeel. Learning interactive real-world simulators. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/c4d66eae503694424123b93ac0fbaf17-Abstract-Conference.html.

Zhaoyang Yang, Yurun Jin, Lizhe Qi, Cong Huang, and Kai Chen. EA-WM: Event-aware generative world model with structured kinematic-to-visual action fields. *arXiv preprint arXiv:2605.06192*, 2026. doi: 10.48550/arXiv.2605.06192. URL https://arxiv.org/abs/2605.06192.

Xianchao Zeng, Xinyu Zhou, Youcheng Li, Jiayou Shi, Tianle Li, Liangming Chen, Lei Ren, and Yong-Lu Li. Diagnose, correct, and learn from manipulation failures via visual symbols. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 42386–42395, 2026. URL https://openaccess.thecvf.com/content/CVPR2026/html/Zeng_Diagnose_Correct_and_Learn_from_Manipulation_Failures_via_Visual_Symbols_CVPR_2026_paper.html.

Haoran Zhang, Yifu Lu, Boyang Wang, Xuhui Kang, Yen-Ling Kuo, Zezhou Cheng, Mengdi Wang, and Odest Chadwicke Jenkins. Foresight: Failure detection for long-horizon robotic manipulation with action-conditioned world model latents. *arXiv preprint arXiv:2606.23085*, 2026. doi: 10.48550/arXiv.2606.23085. URL https://arxiv.org/abs/2606.23085.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 586–595, 2018. URL https://openaccess.thecvf.com/content_cvpr_2018/html/Zhang_The_Unreasonable_Effectiveness_CVPR_2018_paper.html.

Xiangdong Zhang, Jiaqi Liao, Shaofeng Zhang, Fanqing Meng, Xiangpeng Wan, Junchi Yan, and Yu Cheng. VideoREPA: Learning physics for video generation through relational alignment with foundation models. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-4091. URL https://proceedings.nips.cc/paper_files/paper/2025/hash/b1d4973f5d21708abef3cd6f17d842c8-Abstract-Conference.html.

Siyuan Zhou, Yilun Du, Jiaben Chen, Yandong Li, Dit-Yan Yeung, and Chuang Gan. RoboDreamer: Learning compositional world models for robot imagination. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 61885–61896. PMLR, 2024. URL https://proceedings.mlr.press/v235/zhou24f.html.

Fangqi Zhu, Hongtao Wu, Song Guo, Yuxiao Liu, Chilam Cheang, and Tao Kong. IRASim: A fine-grained world model for robot manipulation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9834–9844, 2025. URL https://openaccess.thecvf.com/content/ICCV2025/html/Zhu_IRASim_A_Fine-Grained_World_Model_for_Robot_Manipulation_ICCV_2025_paper.html.

A Appendix

A.1 Failure Data Construction Details

Scope and design axes. Our controlled taxonomy targets physical execution failures, not every possible planning, perception, or system error. AHA/FailGen (Duan et al., 2025) already supports task-specific keyframe changes, configuration sweeps, and simulator success checks. The main-text comparison therefore describes how we organize these shared ideas for action-conditioned future prediction. The substantive axes are phase-conditioned physical modes, semantic applicability before replay, manipulation-primitive intervention, increasing-magnitude search, and separate intervention/outcome validation.

Reading the comparison matrix. Table 1 uses narrowly defined design criteria, not a ranking of dataset quality. *Failure taxonomy* means explicit failure categories; *phase-aware modes* require physically distinguishing similar parameter errors across manipulation stages; *task–fault filtering* means a dedicated applicability decision before simulator search; *primitive injection* means runtime intervention at named manipulation operations; *adaptive severity search* refers to increasing magnitude until a valid failure is found; and *simulator outcome checking* concerns simulator-based validation of the failure-generation stream. A check denotes explicit support, a triangle denotes partial or related support, and a cross denotes that the specifically defined component is not explicitly addressed, not that it is impossible to implement. AHA/FailGen (Duan et al., 2025) already provides phase-related grasp/slip cases, keyframe-level action edits, parameter sweeps, and simulator checking; the triangles distinguish these from a unified phase-conditioned taxonomy, runtime primitive hooks, and a first-valid-failure magnitude search. Guardian (Pacaud et al., 2026) provides subtask-level simulated interventions alongside offline real-data transformations, so partial support must not be read as an absence of simulation or action-level control. REFLECT/RoboFail (Liu et al., 2023) and ViFailback (Zeng et al., 2026) primarily organize failure experiences and annotations rather than the controlled synthesis components compared here. The matrix follows these operational distinctions and does not establish that all real-world execution failures are covered by our taxonomy.

Table 6: Controlled failure modes and their execution stages. These modes apply to the synthetic controlled stream, not an imposed classification of all naturally occurring failures.

Failure mode	Stage	Intervention
Grasp position offset	Grasp	Translate the planned grasp pose
Grasp orientation offset	Grasp	Rotate the planned grasp pose
Weak closure	Grasp/closure	Reduce closure or modify the close command
Target position offset	Placement	Translate the placement target
Target orientation offset	Placement	Rotate the target configuration
Premature release	Transport / pre-placement	Insert an early gripper-opening action
Failure to release	Release	Suppress the intended opening action
Insufficient clearance	Lift/transport	Reduce vertical displacement
Contact point offset	Contact	Displace the contact target
Contact magnitude error	Contact	Scale interaction displacement, strength, or duration
Bimanual phase shift	Coordination	Delay one arm or insert a hold action
Relative pose offset	Handover / coordination	Change the inter-arm relative pose
Timing offset	Action sequencing	Shift the timing of open, close, or contact actions

Applicability. The global taxonomy is human-defined. Task names, descriptions, and successful videos support VLM-assisted selection, yielding a static map $m \mapsto \mathcal{F}_m$. Runtime verification re-

mains authoritative: a proposed fault is not accepted merely because it appears in the map. For example, a pressing task generally does not require a weak-closure fault, whereas inter-arm phase errors require coordinated manipulation.

Increasing-magnitude search. For a fixed task, episode, fault, and sampled variant, let $\mathcal{G} = (\delta_1, \ldots, \delta_K)$ be the tested magnitude sequence. We reset the same scene seed for each trial and stop at the first accepted failure. When the sequence is ordered by magnitude, this is the smallest accepted magnitude among those tested in that sequence; it is not a global optimum across directions, seeds, or all physically possible interventions. The reference normal search allows up to four probe rollouts per fault; repair permits stronger grids and up to eight probes per variant with up to eight variants. Once a probe yields an accepted failure, we replay the same perturbation parameters to save the video and trajectory files. The search is intended to favor informative small deviations, but boundary proximity requires empirical characterization rather than following automatically from acceptance.

Validation and traceability. The acceptance predicates require

$$H_{\text{hook}} \wedge H_{\text{applied}} \wedge H_{\text{valid sim}} \wedge \neg H_{\text{success}}, \tag{6}$$

where the predicates denote an executed hook, an applied fault, a valid simulation, and success under the original task checker, respectively. Rejected attempts are logged separately, including still-successful execution, inapplicable faults, failures before intervention, simulator artifacts, and injector artifacts. These operational checks support intervention attribution within the replay protocol. Each accepted rollout preserves task, source episode, scene seed, fault type, variant/seed, tested magnitudes, executed hooks, and outcome. Generation enumerates applicable task–episode–fault combinations, with coverage reports and targeted repair rather than a fixed number of randomly drawn faults per success. The documented generation set contains 25,000 successful source episodes and 133,500 accepted controlled failures across 50 tasks.

Execution-keyframe definition. The perturbation-execution keyframe records when the deliberately modified action is executed. Together with the fault type and manipulation primitive, it links the intervention to its physical role and subsequent scene evolution.

A.2 SAFF Implementation Details

Architecture and optimized parameters. The video backbone is Wan2.2 TI2V-5B (Wan Team, 2025). KVAFs follow EA-WM (Yang et al., 2026); the implementation encodes their rendered action fields through the VAE and patchifies them onto the video-latent grid. The sparse context extension copies a block at every fifth main DiT index, projects its context features through a linear residual projection, and injects the result before the corresponding main block. At the SAFF stage, the base checkpoint is loaded and fixed, and only new main-DiT LoRA adapters (Hu et al., 2022) receive parameter updates. The reported checkpoints are evaluated after 300,000 optimization steps, as specified in Section 4.1.

Noise convention and conditioning. For the encoded target video z_0, $z_\sigma = (1-\sigma)z_0 + \sigma\epsilon$ uses the decreasing-noise convention at sampling time. The target is $v^\star = \epsilon - z_0$. Here "clean" means noise-free, not a successful reference episode: the target latent comes from the same success or failure video being trained. The initial slot is clamped to its conditioning latent and excluded from future supervision.

Endpoint VJP. For $\hat{z}_0 = z_\sigma - \sigma v_\theta(z_\sigma, \sigma, c)$,

$$J_\sigma = I - \sigma \frac{\partial v_\theta}{\partial z_\sigma}. \tag{7}$$

The feedback-derived correction signal is given in Equation 1, where M excludes the conditioned initial-frame slot and λ_σ is the feedback gain. The reference preconditioner is the identity. The residual is detached when passed as the vector in the vector–Jacobian product; the conditioned slot is zeroed both before and after the VJP. The feedback schedule follows FBFM (Li et al., 2026b),

converted to the noise-level convention:

$$\tau = 1 - \sigma, \qquad r^2 = \frac{\sigma^2}{\tau^2 + \sigma^2}, \qquad \lambda_\sigma = s \min\Big(\frac{\sigma}{\tau r^2}, b\Big), \tag{8}$$

with $s = 1$, $b = 10$, and numerical denominator floors of 10^{-5}. Noise levels are clamped to $[10^{-5}, 1-10^{-5}]$ when evaluating the gain. We compute one endpoint VJP per update. The resulting signal is used only for weights, not to replace the FM velocity target.

DINO semantic-motion objective. Let $\hat{f}^l_{s,j}$ and $f^l_{s,j}$ denote unit-normalized predicted and GT DINOv3 patch features at sampled frame s, patch j, and selected layer l. DINOv3 ViT-B/16 (Siméoni et al., 2025) provides blocks $\{2, 5, 8, 11\}$ in zero-based indexing, after output normalization and removal of special tokens. For each sampled frame pair, the semantic and temporal residuals $a_{s,j}$ and $b_{s,j}$ are defined in Equation 2. The RMS calculation includes an internal 10^{-8} stabilizer. The second quantity is a same-patch feature-change residual, not an explicitly tracked object trajectory. Define spatial 90th-percentile scales $u_s = \max(Q_{.9}(a_s), 10^{-6})$ and $v_s = \max(Q_{.9}(b_s), 10^{-6})$. The candidate map is

$$h_{s,j} = \sqrt{\mathrm{clip}(a_{s,j}/u_s, 0, 1)\,\mathrm{clip}(b_{s,j}/v_s, 0, 1)}, \tag{9}$$

$$c_{s,j} = \mathrm{clip}\big(h_{s,j}/\max(Q_{.9}(h_s), 10^{-6}), 0, 1\big). \tag{10}$$

All scales and candidates are held fixed during differentiation in the auxiliary objective of Equation 3. We sample every fourth RGB frame: 81 video frames yield 21 samples. Gradients from adjacent frame-pair objectives accumulate onto their shared RGB frames, then pass through the frozen VAE decoder to the endpoint latent. The decoder-gradient projection follows Equation 4; the normalized latent gradient defines D in the main text. Tiled decoder VJPs use spatial tile size 12×20 and stride 8×14 with the implementation's overlapping decoder composition. Freezing VAE/DINO parameters does not disable their derivatives with respect to their inputs.

Importance normalization and optimization. The correction map uses channel RMS followed by $3 \times 3 \times 3$ average pooling with replicate padding, then per-sample 90th-percentile normalization over future positions and clipping to $[0, 1]$. The DINO-gradient map uses channel RMS followed by spatial 90th-percentile normalization separately for each future latent time slot. The reference epsilon floor for these importance scales is 10^{-6}. The fused weight is detached and broadcast across channels before computing the normalized weighted MSE. Its raw range is $[1, 2]$; after division by the weight sum, it changes relative gradient shares rather than guaranteeing that every position receives a larger gradient than uniform FM. The two Wan passes share a parameter state and noisy input, with a single optimizer step after the weighted loss. The uniform-FM control bypasses weight construction and uses the same velocity target.

A.3 Benchmark Definitions and Support Rules

Alignment protocol. The reference RoboTwin configuration center-crops both streams to aspect ratio 26:15 before resizing to 224×224. Logical frame indices are aligned at a canonical 15 FPS, then sampled on the GT timeline at 8 FPS. Native aspect-ratio mismatches are retained as audit fields. A shorter prediction repeats its final frame, while the original prediction coverage is retained; extra predicted tail frames are ignored. The supplied configuration starts at time zero and therefore does not automatically exclude an initial conditioning frame. The same crop, frame convention, prompts, and clip selection must be fixed across compared models.

Visual and distributional metrics. PSNR and SSIM (Wang et al., 2004) are computed per RGB frame and averaged; LPIPS (Zhang et al., 2018) uses the AlexNet network and v0.1 calibration. For identical frames, the implementation caps PSNR at 100 dB. With clip_{01} denoting clipping to $[0, 1]$,

$$S_{\mathrm{vis}} = \mathrm{mean}\Big(\mathrm{clip}_{01}(1 - 10^{-\mathrm{PSNR}/20}), \mathrm{clip}_{01}(\mathrm{SSIM}), 1 - \mathrm{clip}_{01}(\mathrm{LPIPS})\Big). \tag{11}$$

FVD (Unterthiner et al., 2018) compares the I3D feature distributions of the full GT and generated collections:

$$\mathrm{FVD} = \|\mu_G - \mu_P\|_2^2 + \mathrm{Tr}\Big(\Sigma_G + \Sigma_P - 2(\Sigma_G\Sigma_P)^{1/2}\Big), \quad S_{\mathrm{dist}} = e^{-\max(\mathrm{FVD},0)/100}. \tag{12}$$

The reference implementation uses one uniformly sampled 16-frame clip per aligned video, skips FVD with fewer than two videos, and warns about limited statistical stability below 256 videos. FVD is not a per-video metric.

Temporal, semantic, and motion metrics. RAFT-Large (Teed & Deng, 2020) estimates GT forward/backward flow. Forward–backward consistency and valid warping coordinates define supported pixels Ω_t. For video X, let $E(X)$ average $|X_t - \mathcal{W}(X_{t+1}, F_t^G)|$ over supported pixels, channels, and valid transitions. We subtract the reference video's own residual:

$$\Delta E = E(P) - E(G), \qquad S_{\text{temp}} = e^{-\max(\Delta E, 0)/0.02}. \tag{13}$$

For semantic similarity, DINOv2-base (Oquab et al., 2023) provides normalized patch features. The object region is the union of GT/predicted motion masks and the scene region is its complement; masks are downsampled to patch weights. This object region is a motion-derived proxy rather than manual semantic segmentation. For video i and region r, let $\mathcal{T}_{i,r}$ contain frames whose region has sufficient patch-weight mass. We first normalize the patch-weighted similarity within each valid frame, then average over valid frames:

$$S_{i,r}^{\text{DINO}} = \frac{1}{|\mathcal{T}_{i,r}|} \sum_{t \in \mathcal{T}_{i,r}} \frac{\sum_j m_{i,t,j}^r \langle f_{i,t,j}^G, f_{i,t,j}^P \rangle}{\sum_j m_{i,t,j}^r}. \tag{14}$$

A region with no valid frame is unavailable rather than assigned zero. Let $\mathcal{R}_i \subseteq \{\text{scene}, \text{object}\}$ be the available regions of video i, and let $\mathcal{I}_{\text{sem}}$ contain videos with at least one available region. The semantic dimension is aggregated in two stages:

$$S_{\text{sem},i} = \frac{1}{|\mathcal{R}_i|} \sum_{r \in \mathcal{R}_i} \text{clip}_{01}\big(S_{i,r}^{\text{DINO}}\big), \qquad S_{\text{sem}} = \frac{1}{|\mathcal{I}_{\text{sem}}|} \sum_{i \in \mathcal{I}_{\text{sem}}} S_{\text{sem},i}. \tag{15}$$

When the object region is unavailable but the scene region is valid, the video contributes its scene score alone; the video itself is not discarded. This aggregation is distinct from averaging the separately reported scene and object dataset means when their valid-video sets differ.

RANSAC homography compensation removes estimated global camera motion before thresholding residual flow at one pixel and applying morphological cleanup. For the resulting spatiotemporal binary masks,

$$S_{\text{motion}} = \frac{|M^G \cap M^P|}{|M^G \cup M^P|}. \tag{16}$$

An empty union yields score 1, signifying no detected motion in either video, not a guarantee of correct physical execution.

Track extraction and association. SAM 3 (Carion et al., 2025) is run independently on GT and generated videos with the same task-specific object prompts and fixed arm prompts. Mask-area thresholds and minimum candidate coverage of 0.10 filter candidates. The reference quality function is

$$Q = 0.34\,\text{coverage} + 0.24\,\text{confidence} + 0.22\,\text{motion} + 0.12\,\text{smoothness} + 0.08\,\text{area stability}. \tag{17}$$

GT candidates are ranked using their own measurements, not predicted trajectory distance. If prediction candidates are absent, the implementation retains the available GT candidate with greatest coverage for the missing-prediction support check. Compatible prediction candidates share the selected GT prompt and are ranked by $0.75\,I + 0.25\,Q$, where I measures position and mask-area agreement on jointly visible frames within the first $\max(3, \lceil 0.25T \rceil)$ frames. The identity term does not minimize full-horizon path error, although Q does use whole-track quality statistics. Coordinates are image-plane mask centroids, not 3D object or end-effector poses.

Confidence and support-aware scoring. For role $r \in \{\text{arm}, \text{object}\}$, let $\mathcal{V}_r$ denote jointly visible frames and $\mathcal{V}_r^{\text{GT}}$ denote GT-visible frames. The normalized centroid error is $d_t = \|c_t^{\text{pred}} - c_t^{\text{GT}}\|_2 / \sqrt{H^2 + W^2}$. Paired detector evidence is $q_t = \min(q_t^{\text{GT}}, q_t^{\text{pred}})$, with localization weight $w_t = \text{clip}(q_t, 0.05, 1)$. We define

$$E_r = \frac{\sum_{t \in \mathcal{V}_r} w_t d_t}{\sum_{t \in \mathcal{V}_r} w_t}, \qquad \bar{q}_r = \frac{1}{|\mathcal{V}_r|} \sum_{t \in \mathcal{V}_r} q_t, \qquad R_r = \frac{|\mathcal{V}_r|}{|\mathcal{V}_r^{\text{GT}}|}, \tag{18}$$

and, for sufficiently supported track pairs,

$$S_r = \exp\left[-\frac{E_r + \beta_{\text{miss}}(1 - R_r) + \beta_{\text{conf}}[q_0 - \bar{q}_r]_+}{s_{\text{traj}}}\right], \tag{19}$$

with $(\beta_{\text{miss}}, \beta_{\text{conf}}, q_0, s_{\text{traj}}) = (0.05, 0.06, 0.60, 0.10)$ and $[u]_+ = \max(u, 0)$. Equations 18–19 use paired SAM3 scores as evidence proxies, with a 0.05 floor only on the per-frame ADE weights. The mean paired confidence itself is not floored at 0.05. The raw alignment omits only the explicit low-confidence penalty; it retains the confidence-weighted ADE and missing-observation cost. FDE is the unweighted position error at the last jointly visible frame and is reported only as a diagnostic. A pair is supported when at least three jointly visible frames cover at least 25% of the aligned sequence. For a missing prediction track or insufficient joint support, a GT coverage of at least 25% yields role score zero; otherwise the role abstains. Missing GT tracks always abstain. The confidence penalty is evaluated only for supported pairs, not by inserting zero-confidence defaults into an otherwise invalid formula.

Dataset aggregation. Available per-video dimension scores are averaged at dataset level. For trajectories, arm and object scores are first averaged separately over their supported or zero-scored samples, then combined with weights 0.4 and 0.6, renormalizing if an entire GT role is unavailable. Let $\mathcal{K}_{\text{avail}}$ be the available dimensions among visual, distributional, temporal, semantic, motion, and trajectory scores. The total is

$$S_{\text{total}} = \frac{100}{|\mathcal{K}_{\text{avail}}|} \sum_{k \in \mathcal{K}_{\text{avail}}} S_k. \tag{20}$$

The complete six-dimensional score uses dataset-level FVD. Single-video runs without FVD are diagnostics and should not be mixed with the formal collection score. The evaluator reports raw and confidence-penalized totals, effective component weights, role support counts, prediction coverage, and track-status counts. Fixed detector confidence is not probability calibration, and the composite fidelity score is not a policy-success probability.

A.4 Additional Metrics

In this simulation comparison, scene scores are available for all 315 videos. The DINO object means use 311, 312, 314, 315, 314, and 311 valid videos for Ctrl-World, DriftWorld, IRASim, WorldGym, ABot-PhysWorld, and Ours, respectively. Videos without a valid object region contribute their scene score alone to the semantic dimension in Equation 15. No DINO value is altered by clipping to $[0, 1]$ in this comparison.

Table 7 reports additional raw video metrics for the same 315-video simulation comparison as Table 2. Pairwise metrics are averaged over available video-level measurements, whereas FVD is computed over the video collection. The benchmark applies per-video normalization and missing-component handling before aggregation, as specified in Appendix A.3; therefore, the composite scores cannot in general be recovered by transforming the rounded averages below. Flow GT is the reference video's warping baseline, and Flow Δ is computed before rounding.

Each generated video contains 81 frames. We select the 17 frames with zero-based indices $0, 5, 10, \ldots, 80$, covering both endpoints and the full video duration, and present them to GPT-5.6 in chronological order. The original full images are preserved without cropping, stretching, or flipping. This visual-judging input protocol is separate from the image alignment used for the video-quality benchmark. All configurations use the same prompt template, with an anonymous evaluation identifier and the task description. The criterion is deliberately lenient: it tests whether the core goal object approximately reaches the requested final spatial relation, rather than requiring a particular arm, exact trajectory, or artifact-free rendering.

The following is an English translation of the fixed Chinese prompt used for evaluation:

> You are a binary visual judge for a robot world model.
>
> Below are 17 frames sampled at fixed intervals from an 81-frame generated video, presented in chronological order.
>
> Anonymous identifier: `{eval_id}`
> Task description: `{task_prompt}`

Table 7: Additional raw video metrics on the held-out simulation set. All models are evaluated on 315 paired videos. Arrows indicate the preferred direction; Flow GT is a shared reference baseline. Best and second-best model results are **bold** and underlined, respectively.

(a) Visual and distributional metrics

Model	n	PSNR ↑	SSIM ↑	LPIPS ↓	FVD ↓
Ctrl-World	315	22.49	0.8603	0.1141	98.1
DriftWorld	315	15.98	0.7306	0.2704	256.5
IRASim	315	12.65	0.6744	0.3387	262.4
WorldGym	315	13.06	0.7305	0.3324	511.8
ABot-PhysWorld	315	13.86	0.7490	0.3223	433.5
Ours	315	**25.10**	**0.8918**	**0.0767**	**55.9**

(b) Temporal, semantic, and motion metrics

Model	Flow pred. ↓	Flow GT	Flow Δ ↓	DINO scene ↑	DINO object ↑	ST-IoU ↑
Ctrl-World	0.01242	0.00769	0.00473	0.7953	**0.7597**	**0.6119**
DriftWorld	0.02588	0.00769	0.01818	0.5435	0.4360	0.3883
IRASim	0.03544	0.00769	0.02775	0.5347	0.3816	0.2339
WorldGym	0.01845	0.00769	0.01075	0.5464	0.3674	0.0541
ABot-PhysWorld	0.03393	0.00769	0.02624	0.5762	0.3931	0.0915
Ours	**0.01089**	0.00769	**0.00320**	**0.8008**	0.7566	0.6025

Table 8: Raw video prediction metrics on the real-world evaluation set. Both models are evaluated on 206 paired videos. Arrows indicate the preferred direction; Flow GT is a shared reference baseline. Best model results are shown in **bold**.

(a) Visual and distributional metrics

Model	n	PSNR ↑	SSIM ↑	LPIPS ↓	FVD ↓
Ctrl-World	206	20.98	0.6016	0.3221	594.7
Ours	206	**24.18**	**0.8659**	**0.0577**	**94.3**

(b) Temporal, semantic, and motion metrics

Model	Flow pred. ↓	Flow GT	Flow Δ ↓	DINO scene ↑	DINO object ↑	ST-IoU ↑
Ctrl-World	0.01441	0.01253	0.00188	0.7241	0.6161	0.4040
Ours	**0.01275**	0.01253	**0.00022**	**0.8905**	**0.7257**	**0.6145**

> Apply a lenient task-completion criterion: judge only whether the core goal object approximately achieves the final spatial relation required by the task. Do not mark the sample as a failure because of the choice of robot arm, the motion trajectory, minor position or orientation deviations, an unclear view of complete gripper release, or minor generation artifacts.

The reported rates therefore characterize the combined world-model and visual-judge pipeline under this outcome-focused criterion, rather than an independent verification of physical correctness.

Table 8 reports the corresponding raw metrics on the same 206 real-world video pairs evaluated in Table 3. Pairwise metrics are averaged over available video-level measurements, whereas FVD is computed over the video collection. Flow GT is the shared reference warping baseline, and Flow Δ is computed before rounding.

Additional trajectory diagnostics. Table 9 reports arm and object ADE, FDE, recall, and paired detection confidence. Role scores in Tables 2 and 3 combine confidence-weighted localization errors, missing-observation costs, and low-confidence penalties under the support rules in Appendix A.3. FDE is a diagnostic only and does not enter the benchmark score. The role scores

cannot in general be reconstructed from the rounded diagnostic means because scoring and support handling precede aggregation.

Table 9: Trajectory diagnostics over available measurements. ADE and FDE are normalized by the image diagonal. Recall and confidence are on $[0, 1]$; confidence is detector evidence, not a calibrated success probability.

	Arm				Object			
Model	ADE ↓	FDE ↓	Recall ↑	Conf.	ADE ↓	FDE ↓	Recall ↑	Conf.
Simulation								
Ctrl-World	0.0673	0.0932	0.9412	0.6905	0.0455	0.1018	0.9247	0.8854
DriftWorld	0.1158	0.1366	0.7639	0.6366	0.0662	0.1416	0.8272	0.8952
IRASim	0.1650	0.2049	0.8461	0.6923	0.0817	0.1747	0.9198	0.8983
WorldGym	0.2065	0.2599	0.8947	0.6607	0.0877	0.1802	0.9888	0.8879
ABot-PhysWorld	0.1737	0.2114	0.8540	0.6981	0.0836	0.1747	0.8121	0.8953
Ours	0.0621	0.0766	0.9154	0.6913	0.0459	0.0891	0.9598	0.8869
Real world								
Ctrl-World	0.1370	0.1459	0.8775	0.6926	0.0723	0.0692	0.8603	0.8455
Ours	0.0439	0.0525	0.9786	0.6886	0.0397	0.0658	0.9459	0.8582

Ablation details. Table 10 reports the normalized dimension scores for the simulation ablations; their equal-weight mean multiplied by 100 gives the total. Tables 11 and 12 provide the corresponding component metrics and trajectory diagnostics.

Table 10: Reported normalized scores for the simulation ablations. The six dimensions lie in $[0, 1]$; Total is on 0–100. Values are shown at their reported precision, with totals rounded to four decimal places.

Variant	Visual	Distrib.	Temporal	Semantic	Motion	Trajectory	Total
w/o both	0.770555	0.136606	0.395666	0.563772	0.190832	0.389468	40.7816
w/o SAFF	0.901233	0.491291	0.852746	0.777835	0.591347	0.639020	70.8912
Ours	0.918170	0.571555	0.862126	0.779410	0.602476	0.656227	73.1661

Table 11: Detailed video metrics for the simulation ablations. Flow GT is a reference residual, not a separate scoring dimension.

(a) Visual and distributional metrics

Variant	n	PSNR ↑	SSIM ↑	LPIPS ↓	FVD ↓
w/o both	315	15.37	0.7537	0.2612	199.1
w/o SAFF	315	22.9943	0.875202	0.094314	71.0719

(b) Temporal, semantic, and motion metrics

Variant	Flow pred. ↓	Flow GT	Flow Δ ↓	DINO scene ↑	DINO object ↑	ST-IoU ↑
w/o both	0.03079	0.00769	0.02309	0.6413	0.4862	0.1908
w/o SAFF	0.011129	0.007694	0.003435	0.800387	0.753802	0.591347

Flow Δ is the difference between prediction and GT warping residuals. FDE is diagnostic only and does not enter the benchmark score. Nonlinear per-video scoring and missing-observation handling precede averaging, so dataset-level diagnostic means do not uniquely determine the normalized dimension scores.

Normalized dimension scores. Table 13 retains the six normalized dimension scores underlying the overall results in Tables 2 and 3. The total is their equal-weight mean multiplied by 100, with per-video normalization and missing-component handling applied as specified in Appendix A.3.

Table 12: Trajectory diagnostics for the simulation ablations. ADE/FDE are normalized by the image diagonal. Recall, confidence, and role scores lie in $[0, 1]$; confidence is detector evidence.

Variant	Role	ADE↓	FDE↓	Recall↑	Conf.	Score↑
w/o both	Arm	0.1398	0.1822	0.8751	0.6837	0.2824
	Object	0.0817	0.1638	0.9299	0.8893	0.4608
w/o SAFF	Arm	0.063870	0.084812	0.910684	0.690887	0.580538
	Object	0.042182	0.086205	0.954420	0.886113	0.678008

When both roles are available, the trajectory dimension combines their dataset-level scores with weights 0.4 for the arm and 0.6 for the object.

Table 13: Normalized dimension scores used for the benchmark totals. All dimension scores are on $[0, 1]$; the overall score is on 0–100. Higher is better.

Model	Visual	Distrib.	Temporal	Semantic	Motion	Trajectory	Overall
Simulation							
Ctrl-World	0.8876	0.3750	0.8122	0.7779	0.6119	0.6061	67.84
DriftWorld	0.7638	0.0769	0.4795	0.4909	0.3883	0.3156	41.92
IRASim	0.6983	0.0725	0.3240	0.4588	0.2339	0.3032	34.84
WorldGym	0.7220	0.0060	0.6174	0.4569	0.0541	0.3174	36.23
ABot-PhysWorld	0.7367	0.0131	0.4087	0.4854	0.0915	0.3149	34.17
Ours	0.9182	0.5716	0.8621	0.7794	0.6025	0.6562	73.17
Real world							
Ctrl-World	0.7295	0.0026	0.9084	0.6701	0.4040	0.4116	52.10
Ours	0.9151	0.3895	0.9609	0.8081	0.6145	0.7107	73.31

Table 14 reports execution-based and estimated success rates for five simulation policies under all four world-model configurations. Figure 5(a,b) compares SAFF with Ctrl-World. The data fractions describe policy training, and all configurations share the same execution-based success rates.

Table 14: Execution-based and world-model-plus-VLM success rates (%) for five simulation policies. The mean row averages the five policy rates equally, rather than pooling evaluation trials.

Policy-training data	Environment execution	Success-only	Standard FM	SAFF (ours)	Ctrl-World
10%	23	79	18	39	47
20%	35	87	21	47	50
30%	42	84	23	43	51
50%	56	89	30	54	68
100%	59	91	29	55	64
Mean	43.0	86.0	24.2	47.6	56.0

Let p_k and $\hat{p}_k$ denote execution-based and visually estimated success rates in percent for policy k, with $K = 5$. In addition to Pearson correlation and Spearman rank correlation, we report

$$\begin{aligned} \mathrm{MAE} &= \frac{1}{K}\sum_{k=1}^{K}|\hat{p}_k - p_k|, \\ \mathrm{Bias} &= \frac{1}{K}\sum_{k=1}^{K}(\hat{p}_k - p_k), \\ E_{\max} &= \max_k |\hat{p}_k - p_k|. \end{aligned} \tag{21}$$

Errors are in percentage points; positive bias indicates overestimation. These statistics compare policy-level rates, not per-video classification accuracy. Table 15 reports the resulting summaries for all four configurations.

Table 15: Summary of all four policy-evaluation configurations. Errors and signed bias are in percentage points. The higher Pearson correlation of standard FM coexists with substantial underestimation.

World model	Pearson r	Spearman ρ	MAE	Bias	Max. error
Success-only	0.904	0.90	43.0	$+43.0$	56
Standard FM	0.983	0.90	18.8	-18.8	30
SAFF (ours)	0.928	0.90	7.0	$+4.6$	16
Ctrl-World	0.926	0.90	13.0	$+13.0$	24

Table 16 reports correlations and success-rate estimation errors for the five real-world policies in Figure 5(c,d). MAE and signed bias average the absolute and signed policy-level errors, respectively, while maximum error is the largest absolute policy-level error. Positive bias indicates overestimation. These statistics describe policy-level rate agreement, not per-video classification accuracy.

Table 16: Agreement between world-model-plus-VLM estimates and real-world execution success rates across five policies. Rate errors are in percentage points (pp). Higher correlations, lower absolute errors, and signed bias closer to zero are better; best results are shown in **bold**.

World model	Pearson $r \uparrow$	Spearman $\rho \uparrow$	MAE (pp) $\downarrow$	Bias (pp; ideal 0)	Max. error (pp) $\downarrow$
Ctrl-World	0.7031	0.7000	16.16	$+10.52$	20.00
SAFF (ours)	**0.9418**	**1.0000**	**8.26**	$\mathbf{+4.34}$	**13.30**

A.5 Additional Qualitative Results

Figure 9 extends the main-text comparison to all evaluated baselines. The predictions exhibit different discrepancies, including limited cube motion in WorldGym, distorted robot geometry in IRASim, and altered cube appearance in ABot-PhysWorld and DriftWorld. Our model better preserves the cube's appearance and post-contact motion relative to the reference sequence.

Figure 7 shows a real-world fruit-to-box sequence. Compared with Ctrl-World, our model more closely reproduces the gripper's approach to the target fruit and its subsequent transport toward the box, although local differences from the reference remain. These examples illustrate interaction fidelity rather than independently establishing task success.



Figure 9: Full qualitative comparison for the simulated sequence shown in Figure 6. From top to bottom: WorldGym, IRASim, ABot-PhysWorld, DriftWorld, Ctrl-World, ours, and ground truth (GT). Time progresses from left to right.