FUTURE ANCHORED VERIFICATION AND ONLINE RECOVERY FOR WORLD ACTION MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

World action models (WAMs) have emerged as a promising paradigm for robotic manipulation. They act by first predicting how a task should be performed and then decoding the actions from that future. However, the remaining actions are invalid once execution drifts from the prediction. Simply replanning from the already out of distribution state rarely restores what the task still requires; existing execution monitors decide when to stop, but not what to restore. We observe that the answer is already in hand: the future the WAM predicted before acting depicts exactly the states it intended to pass through. We introduce FAVOR (**F**uture **A**nchored **V**erification and **O**nline **R**ecovery), a lightweight framework that keeps these predicted frames as anchors and uses them for verification and recovery. An Anchor Verifier compares each observation with its anchor, together with the executed actions, to flag deviations that break the task. Anchor-Guided Recovery then hands the alarmed anchor to a vision-language model, which turns it into a short corrective instruction that WAM could execute under strengthened instruction guidance to get back to the intended future before resuming the task. FAVOR raises the task success of the base WAM from 97.85% to 98.10% on LIBERO and from 72.60% to 72.98% on LIBERO-Plus without modifying the policy.

1 INTRODUCTION

World action models (WAMs) have emerged as a promising paradigm for robotic manipulation. They work in a forward and backward way: they first predict the visual future a task should pass through and then decode the actions that realize the future (Du et al., 2023a; Tian et al., 2025; Wu et al., 2024a; Zhu et al., 2025a; Ye et al., 2026b; Yuan et al., 2026; Li et al., 2026a; Kim et al., 2026; Ye et al., 2026a; Zhou et al., 2026; Zhang et al., 2026b). Unlike policies that map current observations and task instruction directly to actions (Chi et al., 2023; Kim et al., 2024; Black et al., 2025), every action is thus grounded in an explicit future of where the scene should go.

That future is only a plan. Imprecise actions or external disturbances can push the scene off the predicted path, invalidating the remaining actions (Li et al., 2026b; Wang et al., 2026b). Simply replanning from the deviated state, however, offers little help: WAMs are trained on successful demonstrations (Ross et al., 2011; Laskey et al., 2017; Reichlin et al., 2022), so neither the future predicted from such an unfamiliar state nor the actions decoded from it can be trusted. Execution monitors for WAMs help by flagging when the remaining actions should no longer be trusted (Wang et al., 2026b; Liu et al., 2026b; Zhang et al., 2026a; Agia et al., 2025; Xu et al., 2025), but say nothing about what to restore. Recovery methods address this gap, but obtain their target from outside the policy (Dai et al., 2024; Pan et al., 2026). This raises a natural question: can a WAM effectively tell when execution has left its plan, and find its way back on its own?

Our observation is that the target of recovery is already in the policy's hands. The future the WAM predicted depicts the states it intended to pass through, and the actions were generated under its guidance, so the actions remain valid exactly as long as the scene follows it. When a deviation occurs, this prediction still shows the condition that was lost, such as a mug held in the gripper, and that state lies in the region where the policy was trained. A deviation therefore invalidates the remaining actions, but not the states they were meant to establish. This suggests a simple principle: use the policy's own predicted future as an anchor.

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.

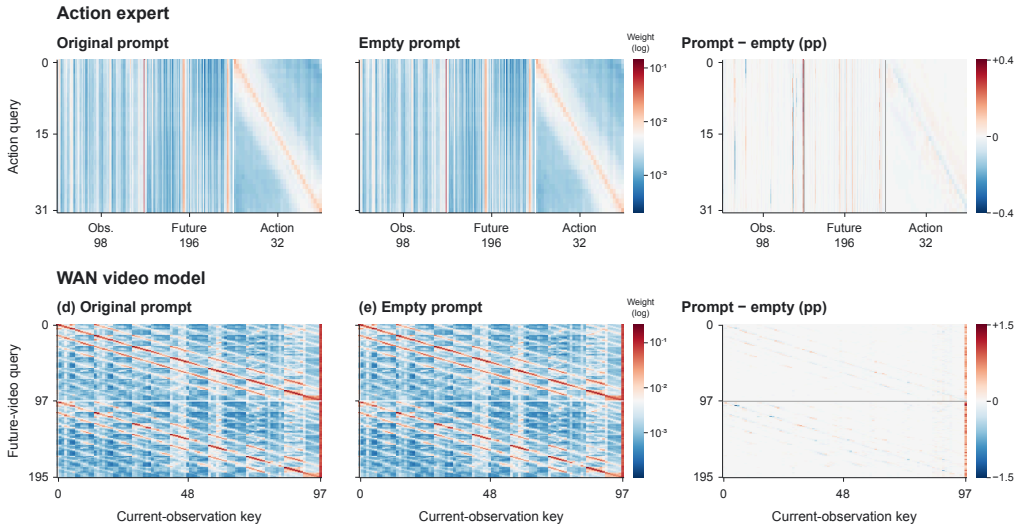

Figure 1: Attention of the frozen WAM under different instructions, with the observation and the random seed fixed. Top: attention of the action tokens. Bottom: attention of the future-video queries over the keys in the video model, under the original prompt, the empty prompt, and their difference. The two conditions produce nearly identical attention maps.

Following this principle, we propose FAVOR (**F**uture **A**nchored **V**erification and **O**nline **R**ecovery), which keeps the predicted frames as anchors and uses them both to verify execution and to guide recovery. Two challenges stand in the way. **(1)** Not every mismatch is a deviation: the anchor is an image produced by a generative model and may contain visual artifacts, so pixel-level differences do not necessarily imply a semantic mismatch. **(2)** The challenge is to convey the anchor to the policy and elicit the intended correction: the anchor is an image, whereas a WAM accepts only an observation and an instruction. Moreover, it leans on visual context and largely ignores the instruction: with the observation fixed, the original and the empty instruction yield nearly identical attention maps (Figure 1). The anchor must therefore be conveyed through a corrective instruction, and the influence of that instruction must be strengthened to elicit the intended correction.

For challenge (1), we introduce the *Anchor Verifier*, which compares each observation with its anchor together with the executed actions and tracks the mismatch along the chunk, so that it responds to deviations that break the task rather than to any visual difference.

For challenge (2), we introduce *Anchor-Guided Recovery* to return the scene to a state the policy knows how to continue from. It hands the alarmed anchor, the current observation, and the task to a vision-language model; the model issues one short instruction that names the condition to restore rather than the pixels to reproduce, and the frozen WAM executes it under strengthened instruction guidance to reach the intended future before deviation.

We evaluate FAVOR on LIBERO (Liu et al., 2023a) and LIBERO-Plus (Fei et al., 2025). FAVOR raises the task success of the base WAM from 97.85% to 98.10% on LIBERO and from 72.60% to 72.98% on LIBERO-Plus, and the Anchor Verifier separates failed from successful episodes with an AUROC of 0.992, above distance-based and learned detectors (Gu et al., 2025; Xu et al., 2025).

In summary, our main contributions are as follows:

- A future-anchored view of WAM execution: the future predicted before acting serves not only as a reference for verifying execution but also as the target that recovery should restore.
- The Anchor Verifier, which grounds execution verification in the WAM's own predicted future and executed actions, identifying when the scene no longer supports continuation.
- The Anchor-Guided Recovery, which translates the alarmed anchor into a corrective instruction and strengthens instruction guidance, enabling a WAM to use its own predicted future as a recovery goal and return to a state from which the original task can continue.

2 Related Work

2.1 World Action Models and Future Prediction

WAMs integrate future prediction into robot control. Joint video–action models couple anticipated visual changes with robot actions in a single generative process (Zhu et al., 2025a; Ye et al., 2026b; Shen et al., 2025; Li et al., 2025; Cen et al., 2025; Zhou et al., 2026; Li et al., 2026a; Ye et al., 2026a; Kim et al., 2026), whereas cascaded interfaces pass visual future states as explicit conditions to inverse dynamics models (IDMs) (Du et al., 2023a; Ko et al., 2024; Black et al., 2023; Hu et al., 2025; Cheang et al., 2024; Tian et al., 2025; Wu et al., 2024a; Liang et al., 2025; Du et al., 2024). The role of prediction nevertheless depends on the inference interface: Fast-WAM separates the benefit of video co-training from that of explicit future generation at test time (Yuan et al., 2026). Related efforts use video models as simulators or reward sources (Yang et al., 2024; Zhu et al., 2025b; Zhou et al., 2024b; Wu et al., 2024b; Escontrela et al., 2023; Jang et al., 2025; Liao et al., 2026; Guo et al., 2026; Assran et al., 2025), learn latent actions from video (Ye et al., 2025), and diagnose whether generated futures actually follow the commanded actions (Chen et al., 2026; Liu et al., 2026a; Li et al., 2026b). We focus on WAMs in which predicted future states condition action generation, so that the intended state changes are available alongside the resulting actions.

In these interfaces, and in work on aligning visual features and instructions with action decoding (Qiu et al., 2026; He et al., 2026; Liu et al., 2026c), the predicted future is an input to action generation and plays no further role once the actions are produced. FFDC is the exception: it retains the prediction and compares it with incoming observations to judge whether the remaining actions can still be trusted, shortening execution or replanning when they cannot (Wang et al., 2026b). HarnessWAM also intervenes during WAM execution, but monitors subgoal completion rather than the prediction (Gu et al., 2026). Here, as in adaptive-horizon methods (Wang et al., 2026b;a; Nie et al., 2026), the signal only decides whether to continue, and replanning starts from the deviated state.

2.2 Execution Verification and Recovery

Execution monitoring for learned policies draws on internal features, action consistency, and semantic progress (Gu et al., 2025; Agia et al., 2025; Xu et al., 2025; Du et al., 2023b; Duan et al., 2025; Thoduka et al., 2025; Sliwowski & Lee, 2025; Sudhakar et al., 2026; Zhao et al., 2026; Liu et al., 2026b; Zhang et al., 2026a; Elhafsi et al., 2023; Sinha et al., 2024; Ren et al., 2023; Ma et al., 2025), but a detector says only that something went wrong, not what state to return to. Existing recovery methods supply that target from outside the acting policy. Planner-level loops revise behavior from environmental feedback and constraint violations (Huang et al., 2023; Guo et al., 2023; Zhou et al., 2024a); test-time language correction takes the instruction from a human (Liu et al., 2023c;b; Shi et al., 2024); policy-level recovery for VLAs is learned from failure-and-recovery data (Liu et al., 2023e; Dai et al., 2024; Li et al., 2024; Lu et al., 2025) or inferred by an auxiliary predictive component, whether a dynamics model (Pan et al., 2026) or imagined states toward which recovery is directed (Shin et al., 2026; Zhang et al., 2026c). Each obtains its objective at the cost of training or a second model, and none derives it from the plan the policy was executing. FAVOR takes the objective from that plan: the future the WAM predicted before acting is what recovery restores, and the VLM only translates it into an instruction rather than estimating a target of its own.

3 Preliminaries

World action models. A WAM first predicts the visual states the task should pass through and then inverts them into actions (Du et al., 2023a; Tian et al., 2025; Yuan et al., 2026; Zhang et al., 2026b); we build on the IDM variant of Fast-WAM (Yuan et al., 2026). Formally, let o_t denote the observation at step t, and ℓ the task instruction. At each planning step t_c the policy produces a chunk of H actions $\mathbf{A}_c = (a_{c,0}, \ldots, a_{c,H-1})$, with $a_{c,k}$ applied at step $t_c + k$, and the WAM factorizes the chunk through the future $\mathbf{Z}_c$, a sequence of visual states over the chunk horizon:

$$p(\mathbf{A}_c \mid o_{t_c}, \ell) = \int \underbrace{p(\mathbf{Z}_c \mid o_{t_c}, \ell)}_{\text{future prediction}} \underbrace{p(\mathbf{A}_c \mid o_{t_c}, \mathbf{Z}_c, \ell)}_{\text{inverse dynamics}} \, \mathrm{d}\mathbf{Z}_c. \tag{1}$$

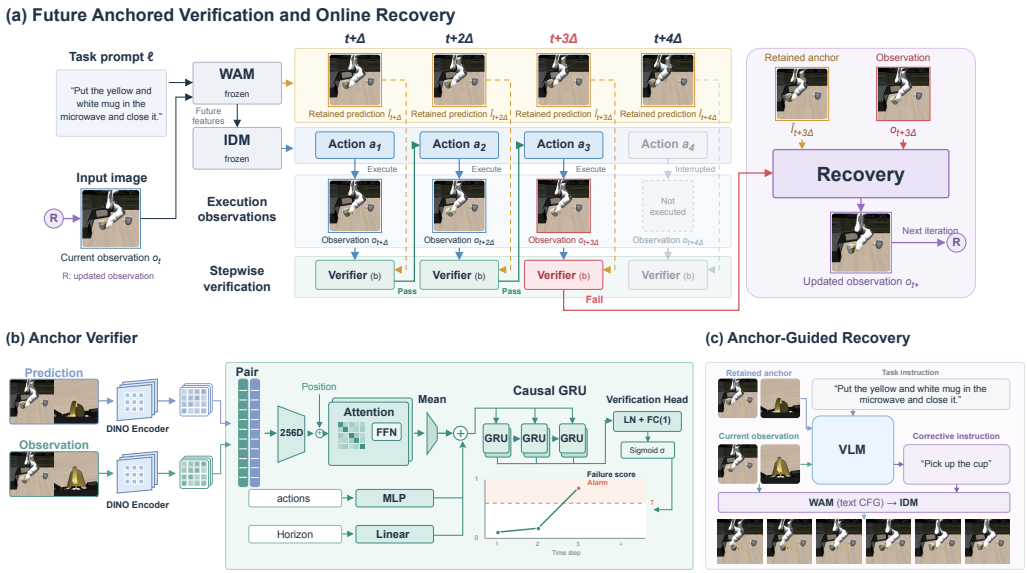

Figure 2: Overview of FAVOR. **(a)** The frozen WAM predicts future frames and the IDM decodes an action chunk; the predicted frames are retained as anchors. Every Δ steps, the verifier compares the observation with its anchor: a pass continues execution, and the failure interrupts the remaining actions and triggers recovery. **(b)** The Anchor Verifier encodes the prediction–observation pair with DINO, fuses it with action and horizon embeddings, and outputs a failure score through a causal GRU; an alarm is raised when the score exceeds τ. **(c)** Anchor-Guided Recovery: a VLM reads the alarmed anchor, the current observation, and the task, and issues a corrective instruction.

Future prediction. The video model is a diffusion transformer (Peebles & Xie, 2023; Team Wan et al., 2025) trained by flow matching (Lipman et al., 2023). Let z_t be the latent of I_t and $\mathbf{Z}_c = z_{t_c+\Delta:t_c+H}$ the latents of the chunk's future frames. Flow matching learns a velocity field that transports Gaussian noise $\mathbf{Z}^{(1)} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ to the future $\mathbf{Z}^{(0)} = \mathbf{Z}_c$ through the ordinary differential equation

$$\frac{\mathrm{d}\mathbf{Z}^{(s)}}{\mathrm{d}s} = \mathbf{v}_\theta\big(\mathbf{Z}^{(s)}, s \mid z_{t_c}, \ell\big), \qquad s \in [0,1], \tag{2}$$

conditioned on the current latent and the instruction. Rectified flow (Liu et al., 2023d) takes the straight path $\mathbf{Z}^{(s)} = (1-s)\,\mathbf{Z}_c + s\,\mathbf{Z}^{(1)}$, so the target velocity is simply $\mathbf{Z}^{(1)} - \mathbf{Z}_c$:

$$\mathbf{v}_\theta\big(\mathbf{Z}^{(s)}, s \mid z_{t_c}, \ell\big) \approx \mathbf{Z}^{(1)} - \mathbf{Z}_c. \tag{3}$$

At inference, the future is obtained by integrating Eq. 2 backward from noise to a clean latent,

$$\mathbf{Z}^{(s_{i-1})} = \mathbf{Z}^{(s_i)} - (s_i - s_{i-1})\,\mathbf{v}_\theta\big(\mathbf{Z}^{(s_i)}, s_i \mid z_{t_c}, \ell\big), \qquad i = T, \dots, 1, \tag{4}$$

which yields the predicted future $\widehat{\mathbf{Z}}_c = \mathbf{Z}^{(s_0)}$. Decoding $\widehat{\mathbf{Z}}_c$ produces H/Δ future frames $(\hat{I}_{c,1}, \dots, \hat{I}_{c,H/\Delta})$, spaced Δ control steps apart. Each frame $\hat{I}_{c,j}$ depicts the state expected at step $t_c + j\Delta$, with $\hat{I}_{c,0} = I_{t_c}$ denoting the current observation. These predicted frames serve as anchors for execution verification and recovery in Section 4.

Inverse dynamics. Given the current latent and the predicted future, the IDM infers the actions needed to realize that future (Yuan et al., 2026). The action tokens are denoised with the same flow-matching procedure, conditioned on the current latent, the predicted future, and the task instruction,

$$\mathbf{A}_c \sim p(\cdot \mid z_{t_c}, \widehat{\mathbf{Z}}_c, \ell). \tag{5}$$

Specifically, the predicted frames and generated actions follow a strict temporal correspondence: each group of Δ consecutive actions is intended to realize the transition between two consecutive predicted frames,

$$\hat{I}_{c,j-1} \xrightarrow{\mathbf{a}_{c,j}} \hat{I}_{c,j}, \qquad \mathbf{a}_{c,j} = (a_{c,(j-1)\Delta}, \dots, a_{c,j\Delta-1}), \qquad j = 1, \dots, H/\Delta, \tag{6}$$

with $\hat{I}_{c,0} = I_{t_c}$ for the first action. The Action $\mathbf{a}_{c,j}$ is executed from step $t_c + (j-1)\Delta$ through step $t_c + j\Delta - 1$, so that $\hat{I}_{c,j}$ describes the state these actions are meant to reach at step $t_c + j\Delta$. At that point, the actual observation $I_{t_c+j\Delta}$ can be compared with $\hat{I}_{c,j}$ to assess whether execution has reached the intended state.

4 METHOD

Overview. Motivated by the observation in Section 1 that the future predicted before acting depicts the states the chunk was meant to reach, and that these states lie where the policy is reliable, FAVOR therefore keeps the predicted frames as *anchors* and uses them twice (Figure 2): an Anchor Verifier checks whether the current observation has deviated too far from its anchor, and Anchor-Guided Recovery uses that anchor to guide the robot back on course.

4.1 ANCHOR VERIFIER

Whether an observation still matches its anchor $\hat{I}_{c,j}$ need be judged semantically and over time; a fixed image distance (Wang et al., 2026b) captures neither, so we learn the verifier from where the mismatch lies, how much motion the actions $\mathbf{a}_{c,j}$ command, and how it evolves along the chunk.

Verifier Architecture. As shown in Figure 2(b), at anchor j of chunk c a frozen DINOv2 encoder (Oquab et al., 2024) tokenizes the observation $I_{t_c+j\Delta}$ and the anchor $\hat{I}_{c,j}$ into $O_{c,j}$ and $\hat{O}_{c,j}$, which together with the executed actions $\mathbf{a}_{c,j}$ and the anchor horizon $p_j = j\Delta/H$ form the input

$$X_{c,j} = \big[\, O_{c,j},\ \hat{O}_{c,j},\ \mathbf{a}_{c,j},\ p_j \,\big]. \tag{7}$$

Observation and anchor tokens are paired position by position and fused by a spatial Transformer into a visual summary, to which embeddings of the actions and of the position are added, giving the anchor feature $x_{c,j}$. A GRU (Cho et al., 2014) carries these features across the anchors of the chunk and a linear head g produces the logit $u_{c,j}$ and the deviation score $s_{c,j}$

$$u_{c,j} = g\big(\mathrm{GRU}(x_{c,1}, \ldots, x_{c,j})\big), \qquad s_{c,j} = \sigma(u_{c,j}) \in [0,1]. \tag{8}$$

An alarm is raised when $s_{c,j}$ exceeds a threshold τ fixed before evaluation.

Training data. To train the Anchor Verifier, we construct two datasets with different levels of supervision. The first is a *locally labeled* set that carries a deviation label at every anchor: from the same simulator state we execute a clean and a perturbed version of the planned actions, and label each anchor by whether the perturbed execution has departed from the clean one. We use two perturbations, late Gaussian noise added to the arm commands in the latter half of the chunk and tail scaling that multiplies these commands by a fixed factor. The second is an *episode-labeled* set that carries only a terminal success or failure label per episode; it contains natural rollouts of the policy together with the demonstrations from its training set, which all count as successes.

Training strategy. We initialize the verifier using local supervision and then adapt it to natural rollouts using terminal outcomes. Since a failed rollout may contain many anchors that were correctly reached, we use multiple-instance learning to supervise episode-level predictions. Let $u_{i,c,j}$ denote the verifier logit at anchor (c,j) of episode i, $\mathcal{V}_i$ the set of its valid anchors, and $m_i = \max_{(c,j)\in\mathcal{V}_i} u_{i,c,j}$ the episode logit. For the failed and successful episode sets $\mathcal{F}$ and $\mathcal{S}$ in each training batch, we minimize

$$L_F = \frac{1}{|\mathcal{F}|} \sum_{i\in\mathcal{F}} \mathrm{sp}(-m_i), \tag{9}$$

$$L_S = \frac{1}{|\mathcal{S}|} \sum_{i\in\mathcal{S}} \left[\frac{1}{2}\,\mathrm{sp}(m_i) + \frac{1}{2|\mathcal{V}_i|} \sum_{(c,j)\in\mathcal{V}_i} \mathrm{sp}(u_{i,c,j}) \right], \tag{10}$$

$$L_{\mathrm{ep}} = \frac{1}{2}(L_F + L_S), \qquad \mathrm{sp}(x) = \log(1 + e^x). \tag{11}$$



Figure 3: Effect of the guidance scale on recovery predictions. All branches start from the same trigger state and the corrective instruction "pick up the cup".

The failure term encourages at least one high score anchor without assigning positive targets throughout a failed episode, while the success term suppresses both the strongest and the average score of successful episodes, so that alarms stay rare while execution is on track.

4.2 Anchor-Guided Recovery

At the alarm, the controller discards the unexecuted actions of the chunk and keeps the alarmed anchor I^{anc}. The WAM is conditioned on a current observation and an instruction, so the anchor needs to reach it through language. Specifically, we use a VLM $\mathcal{M}$ to generate a corrective instruction for transitioning from the current observation $I_{t^\star}$ to the anchored state I^{anc}.

$$\ell_{\mathrm{corr}} = \mathcal{M}(I^{\mathrm{anc}}, I_{t^\star}, \ell). \tag{12}$$

The anchor specifies the nominal state that execution was expected to reach. Expressing this target as a short instruction makes the correction depend on the corresponding object-level state change, such as grasping an object, rather than on exact visual agreement with the predicted frame. If the VLM judges that no correction is needed, it returns the original task, $\ell_{\mathrm{corr}} = \ell$. The WAM then generates futures and actions from the current observation and ℓ_{corr} using the same inference procedure as in normal execution.

Recovery starts from a state outside the policy's training distribution under an instruction it never saw, and, as Figure 1 shows, the instruction has little influence in WAM; we therefore apply classifier-free guidance (CFG) (Ho & Salimans, 2022) during recovery to strengthen the influence of ℓ_{corr} on the predicted future:

$$\hat{\mathbf{v}}_\theta\big(\mathbf{Z}^{(s)}, s \mid z_{t^\star}, \ell_{\mathrm{corr}}; \gamma\big) = \mathbf{v}_\theta(\cdot \mid z_{t^\star}, \varnothing) + \gamma\big[\mathbf{v}_\theta(\cdot \mid z_{t^\star}, \ell_{\mathrm{corr}}) - \mathbf{v}_\theta(\cdot \mid z_{t^\star}, \varnothing)\big], \tag{13}$$

where $\varnothing$ denotes the empty instruction and γ the guidance scale (Figure 3). The IDM then decodes the guided future into recovery actions.

At the end of the window, the VLM compares the current observation with the anchor once more: if it judges that the anchored condition is restored, control returns to the policy.

Table 1: Task success (%, ↑) on LIBERO and LIBERO-Plus, and the LIBERO failure rate (%, ↓) with its relative change from the base WAM. Bold marks the best entry in each column; the last row gives the gain of FAVOR over the base WAM in percentage points.

	LIBERO							LIBERO-Plus
	Success (%, ↑)					Failure (%, ↓)		Success (↑)
Method	Spatial	Object	Goal	Long	Avg.	Rate	Rel.	Avg.
Fast-WAM-IDM	98.20	98.90	98.30	96.00	97.85	2.15	–	72.60
+ Verifier, direct replanning	**98.30**	99.20	98.20	96.00	97.92	2.08	−3.3%	72.60
+ Verifier, correction w/o anchor	**98.30**	99.30	**98.40**	96.10	98.02	1.98	−7.9%	72.53
FAVOR (anchor-guided recovery)	**98.30**	**99.40**	**98.40**	**96.30**	**98.10**	**1.90**	**−11.6%**	**72.98**
Gain over base (pp)	+0.10	+0.50	+0.10	+0.30	+0.25	−0.25		+0.37

5 EXPERIMENTS

5.1 EXPERIMENTAL SETUP

Benchmarks and evaluation protocols. We evaluate FAVOR on LIBERO (Liu et al., 2023a) and LIBERO-Plus (Fei et al., 2025). LIBERO comprises four suites of ten tasks each: Spatial, Object, Goal, and Long (LIBERO-10). LIBERO-Plus uses the LIBERO tasks while varying the environment, task configuration, and observation distribution beyond the standard setting. We run 100 rollouts per task on LIBERO and a total of 1698 rollouts on LIBERO-Plus.

Implementation details. We instantiate FAVOR on the IDM variant of Fast-WAM (Yuan et al., 2026). The Anchor Verifier is trained on a single NVIDIA A5000 GPU with 24 GB of memory. We use Qwen3-VL-2B-Instruct (Bai et al., 2025) as the VLM in Anchor-Guided Recovery. All evaluations are performed on four A5000 GPUs.

Evaluation metrics. The primary metric is native task success, determined by the environment's own success condition and computed over the complete evaluation cohort, including episodes in which no intervention occurs, so that it credits successful recoveries and penalizes interventions that disrupt otherwise successful execution. The Anchor Verifier and Anchor-Guided Recovery are additionally evaluated in isolation in Section 5.3.

5.2 MAIN RESULTS

Results on LIBERO and LIBERO-Plus. Table 1 reports native task success. FAVOR raises the average success of the base WAM from 97.85% to 98.10% on LIBERO, cutting the failure rate by 11.6% relative, and from 72.60% to 72.98% on LIBERO-Plus. Both components are needed. From the same alarms, direct replanning under the original instruction removes only 3.3% of the base failures on LIBERO and none on LIBERO-Plus, and language correction without the anchor removes 7.9% on LIBERO but slightly hurts on LIBERO-Plus (72.53%); supplying the anchor accounts for the rest. The gain is largest on Object, whose tasks share one scene and differ only in the target object, so the policy relies more on the instruction and a corrective instruction moves it more, and on LIBERO-10, where an early deviation has the most time to compound.

Qualitative results. Figure 4 shows two cases that fail under native continuation and are rescued by FAVOR. In both, the Anchor Verifier detects the deviated state as soon as the observation stops matching its anchor: a bottle that the plan expects in the gripper but is not, and a mug that has left its intended path. Anchor-Guided Recovery reads the anchor to identify what was lost and issues a short instruction that restores it, "pick up the orange juice bottle" and "take the white mug"; the scene returns to the anchored state, the original plan applies again, and both tasks complete. Native continuation, by contrast, keeps executing the remaining actions as if nothing had happened, and both tasks fail once the budget runs out.

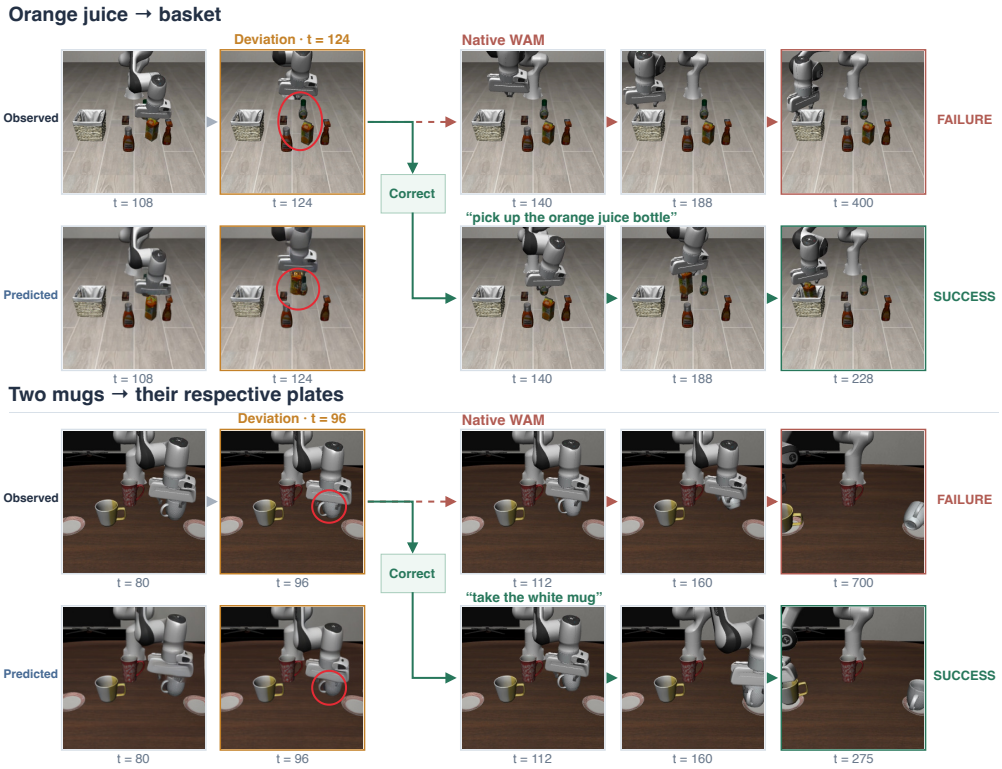

Figure 4: Qualitative examples of Anchor-Guided Recovery on LIBERO. In each case, the top row shows the observed execution and the bottom row the WAM's retained prediction up to the alarm (orange frame); red circles mark where the observation departs from its anchor. Right of the alarm, native continuation under the original instruction fails (top), while the corrective instruction issued by the VLM (green) restores the anchored condition and the task succeeds (bottom). Both branches start from the same simulator state at the alarm.

5.3 Ablation Studies

Anchor Verifier. We compare the Anchor Verifier with DINO adaptations of SAFE-MLP and SAFE-LSTM (Gu et al., 2025), and FAIL-Detect logpZO (Xu et al., 2025), as well as DINO feature distance based on DINOv2 (Oquab et al., 2024) and PSNR (Huynh-Thu & Ghanbari, 2008). All six methods score the same saved base-WAM test episodes, whose initial states are disjoint from those used to train, validate, and calibrate the learned detectors.

Each episode is scored by its maximum anomaly score over the anchors of complete chunks. The baselines split into learned detectors that do not see the anchor and fixed comparisons with it. SAFE-MLP and SAFE-LSTM (Gu et al., 2025) learn a failure classifier from terminal labels on the same DINO features of the observation, per anchor and over the anchor sequence respectively; FAIL-Detect logpZO (Xu et al., 2025) fits a flow model to the observation features of successful episodes and scores an episode by how unlikely its features are under that model. DINO distance averages Euclidean differences between corresponding observed and anchor tokens, and PSNR uses the mean squared RGB error between the observation and the anchor across both cameras, negated so that higher scores indicate greater discrepancy; neither is trained. Table 2 reports episode-level AUROC and AUPRC with task failure as the positive class, and failure coverage at each detector's own threshold, calibrated on successful episodes at a 5% alarm rate and fixed before testing.

The Anchor Verifier leads on all three metrics. The fixed comparisons rank episodes well (AUROC 0.94–0.96) but have low AUPRC and coverage, so a raw distance to the anchor cannot tell a lost grasp from a harmless mismatch; the learned detectors without the anchor do no better, and SAFE-LSTM

Table 2: Detector comparison on LIBERO.

Detector	AUROC ↑	AUPRC ↑	Coverage ↑
SAFE-MLP (DINO) (Gu et al., 2025)	0.935	0.664	67.6%
SAFE-LSTM (DINO) (Gu et al., 2025)	0.712	0.152	38.2%
FAIL-Detect logpZO (DINO) (Xu et al., 2025)	0.951	0.512	73.5%
DINO feature distance (Oquab et al., 2024)	0.944	0.401	73.5%
PSNR (Huynh-Thu & Ghanbari, 2008)	0.956	0.472	67.6%
Anchor Verifier (ours)	0.992	0.781	97.1%

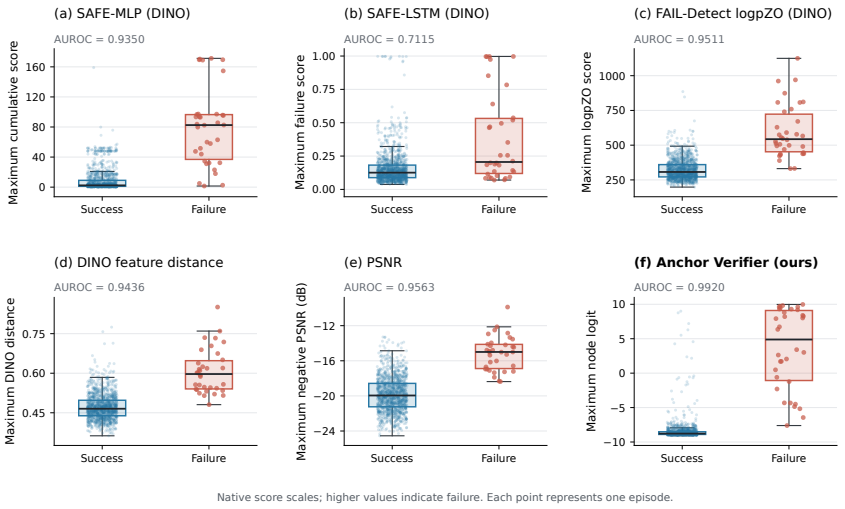

Figure 5: Distribution of the maximum episode score for successful and failed LIBERO episodes under each detector, on its native scale (higher indicates failure); each point is one episode.

collapses. Learning the comparison against the anchor combines the two and raises AUPRC by 0.12 and coverage by 24 points. Figure 5 shows why: under every baseline the scores of successful episodes spread over the same range as those of failures, whereas under the Anchor Verifier the successes collapse into a tight band at the bottom of the scale and the failures rise well above it, so one threshold separates them, which is what yields high coverage at a low alarm rate.

Anchor-Guided Recovery. We ablate the guidance scale of Anchor-Guided Recovery: Figure 3 fixes the state and the corrective instruction "pick up the cup" and varies only γ in Eq. 13. At $\gamma = 1$, which reduces to plain conditional sampling, the predicted future is nearly identical to the continuation under the original task: the correction is ignored. At $\gamma = 3$ the prediction begins to depart from that continuation, but it does not yet realize the instruction, as the arm leaves the original path without reaching a grasp; the correction is registered but not followed. At $\gamma = 5$ the predicted future shows the arm grasping the cup, and the IDM turns this future into the corresponding recovery actions. The guidance scale thus sets how far the corrective instruction overrides the policy's habitual continuation, and it must be large enough for the instruction to prevail.

6 Conclusion

Conclusion World action models plan by predicting the future, yet once execution drifts from that plan they have no way to notice or to recover. FAVOR adds this capability to a frozen WAM by reusing its own prediction: the Anchor Verifier compares observations with their anchors to detect deviations that break the task, and Anchor-Guided Recovery turns the alarmed anchor into a corrective instruction that steers the policy back to a state it can continue from. On LIBERO and LIBERO-Plus it reduces failures and improves end-to-end success without modifying the policy.

AI USE STATEMENT

In this work, we used generative AI tools to edit the paper for readability, to suggest the structure of sections, and to format references. We have reviewed all AI-assisted work: every edited passage was checked by the authors against the underlying results, and all claims, numbers, and citations were verified manually. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

The appendix, provided as supplementary material, contains the details needed to reproduce our results: Appendix A gives the complete controller as pseudocode, Appendix B lists the default settings and the training procedure of the Anchor Verifier, and Appendix C reproduces the exact prompt given to the VLM together with its decoding settings. All experiments use the public LIBERO and LIBERO-Plus benchmarks and the released Fast-WAM checkpoints.

REFERENCES

Christopher Agia, Rohan Sinha, Jingyun Yang, Ziang Cao, Rika Antonova, Marco Pavone, and Jeannette Bohg. Unpacking failure modes of generative policies: Runtime monitoring of consistency and progress. In *Proceedings of The 8th Conference on Robot Learning*, volume 270 of *Proceedings of Machine Learning Research*, pp. 689–723. PMLR, 06–09 Nov 2025. URL https://proceedings.mlr.press/v270/agia25a.html.

Mahmoud Assran, Adrien Bardes, David Fan, Quentin Garrido, Russell Howes, Mojtaba Komeili, Matthew Muckley, Ammar Rizvi, Claire Roberts, Koustuv Sinha, Artem Zholus, Sergio Arnaud, Abha Gejji, Ada Martin, Francois Robert Hogan, Daniel Dugas, Piotr Bojanowski, Vasil Khalidov, Patrick Labatut, Francisco Massa, Marc Szafraniec, Kapil Krishnakumar, Yong Li, Xiaodong Ma, Sarath Chandar, Franziska Meier, Yann LeCun, Michael Rabbat, and Nicolas Ballas. V-JEPA 2: Self-supervised video models enable understanding, prediction and planning. *arXiv preprint arXiv:2506.09985*, 2025. URL https://arxiv.org/abs/2506.09985.

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

Kevin Black, Mitsuhiko Nakamoto, Pranav Atreya, Homer Walke, Chelsea Finn, Aviral Kumar, and Sergey Levine. Zero-Shot Robotic Manipulation with Pretrained Image-Editing Diffusion Models. *arXiv preprint arXiv:2310.10639*, 2023. doi: 10.48550/arXiv.2310.10639. URL https://arxiv.org/abs/2310.10639.

Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Robert Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Lucy Xiaoyang Shi, Laura Smith, James Tanner, Quan Vuong, Anna Walling, Haohuan Wang, and Ury Zhilinsky. π_0: A vision-language-action flow model for general robot control. In *Proceedings of Robotics: Science and Systems*, LosAngeles, CA, USA, June 2025. doi: 10.15607/RSS.2025.XXI.010. URL https://www.roboticsproceedings.org/rss21/p010.html.

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

Chi-Lam Cheang, Guangzeng Chen, Ya Jing, Tao Kong, Hang Li, Yifeng Li, Yuxiao Liu, Hongtao Wu, Jiafeng Xu, Yichu Yang, Hanbo Zhang, and Minzhao Zhu. GR-2: A generative video-language-action model with web-scale knowledge for robot manipulation. *arXiv preprint arXiv:2410.06158*, 2024. URL https://arxiv.org/abs/2410.06158.

Sixiang Chen, Jiaming Liu, Jixian Wu, Yichen Guo, Tinghao Wang, Siyuan Qian, Hao Chen, Jiajun Cao, Jian Tang, and Shanghang Zhang. Do robotic world models really follow actions? diagnosing and aligning action-conditioned generation for policy learning. *arXiv preprint arXiv:2608.24885*, 2026.

Cheng Chi, Siyuan Feng, Yilun Du, Zhenjia Xu, Eric Cousineau, Benjamin CM Burchfiel, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. In *Proceedings of Robotics: Science and Systems*, Daegu, Republic of Korea, July 2023. doi: 10.15607/RSS.2023.XIX.026. URL https://roboticsproceedings.org/rss19/p026.html.

Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1724–1734, Doha, Qatar, oct 2014. Association for Computational Linguistics. doi: 10.3115/v1/D14-1179. URL https://aclanthology.org/D14-1179/.

Yinpei Dai, Jayjun Lee, Nima Fazeli, and Joyce Chai. RACER: Rich Language-Guided Failure Recovery Policies for Imitation Learning. *arXiv preprint arXiv:2409.14674*, 2024. doi: 10.48550/arXiv.2409.14674. URL https://arxiv.org/abs/2409.14674.

Yilun Du, Sherry Yang, Bo Dai, Hanjun Dai, Ofir Nachum, Josh Tenenbaum, Dale Schuurmans, and Pieter Abbeel. Learning universal policies via text-guided video generation. In *Advances in Neural Information Processing Systems*, volume 36, pp. 9156–9172. Curran Associates, Inc., 2023a. doi: 10.52202/075280-0403. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/1d5b9233ad716a43be5c0d3023cb82d0-Paper-Conference.pdf.

Yilun Du, Mengjiao Yang, Pete Florence, Fei Xia, Ayzaan Wahid, Brian Ichter, Pierre Sermanet, Tianhe Yu, Pieter Abbeel, Joshua B. Tenenbaum, Leslie Kaelbling, Andy Zeng, and Jonathan Tompson. Video language planning. In *International Conference on Learning Representations*, 2024.

Yuqing Du, Ksenia Konyushkova, Misha Denil, Akhil Raju, Jessica Landon, Felix Hill, Nando de Freitas, and Serkan Cabi. Vision-language models as success detectors. *arXiv preprint arXiv:2303.07280*, 2023b.

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://github.com/NVlabs/AHA.

Amine Elhafsi, Rohan Sinha, Christopher Agia, Edward Schmerling, Issa Nesnas, and Marco Pavone. Semantic anomaly detection with large language models. *Autonomous Robots*, 47:1035–1055, 2023. URL https://arxiv.org/abs/2305.11307.

Alejandro Escontrela, Ademi Adeniji, Wilson Yan, Ajay Jain, Xue Bin Peng, Ken Goldberg, Youngwoon Lee, Danijar Hafner, and Pieter Abbeel. Video prediction models as rewards for reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 36, pp. 68760–68783. Curran Associates, Inc., 2023. doi: 10.52202/075280-3009. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/d9042abf40782fbce28901c1c9c0e8d8-Paper-Conference.pdf.

Senyu Fei, Siyin Wang, Junhao Shi, Zihao Dai, Jikun Cai, Pengfang Qian, Li Ji, Xinzhe He, Shiduo Zhang, Zhaoye Fei, Jinlan Fu, Jingjing Gong, and Xipeng Qiu. LIBERO-Plus: In-depth robustness analysis of vision-language-action models. *arXiv preprint arXiv:2510.13626*, 2025. URL https://arxiv.org/abs/2510.13626.

Qiao Gu, Yuanliang Ju, Shengxiang Sun, Igor Gilitschenski, Haruki Nishimura, Masha Itkina, and Florian Shkurti. SAFE: Multitask failure detection for vision-language-action models. In *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 40041–40076. Curran Associates, Inc., 2025. doi: 10.52202/085713-1337. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/392d0d05e2f514063e6ce6f8b370834c-Paper-Conference.pdf.

Zhaopeng Gu, Bingke Zhu, Tianxi Lin, Guibo Zhu, Yingying Chen, Kai Wang, Tingyu Yuan, Chaoyang Zhao, Zhaowen Li, Peng Su, and Jinqiao Wang. HarnessWAM: Bridging Prediction and Deliberation in World Action Models. *arXiv preprint arXiv:2608.09516*, 2026. doi: 10.48550/arXiv.2608.09516. URL https://arxiv.org/abs/2608.09516.

Yanjiang Guo, Yen-Jen Wang, Lihan Zha, and Jianyu Chen. DoReMi: Grounding language model by detecting and recovering from plan-execution misalignment. *arXiv preprint arXiv:2307.00329*, 2023.

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 (ICLR)*, 2026. URL https://github.com/Robert-gyj/Ctrl-World.

Junjie He, Junfeng Li, Zhide Zhong, Haodong Yan, Ruixin Li, Yangyang Zheng, Jiaguan Zhu, Tianran Zhang, Yuqiao Du, Wen Chen, Shunbo Zhou, and Haoang Li. SG-WAM: Text-Grounded and Spatial-aware Semantic Guidance for World-Action Models. *arXiv preprint arXiv:2608.08839*, 2026. doi: 10.48550/arXiv.2608.08839. URL https://arxiv.org/abs/2608.08839.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

Yucheng Hu, Yanjiang Guo, Pengchao Wang, Xiaoyu Chen, Yen-Jen Wang, Jianke Zhang, Koushil Sreenath, Chaochao Lu, and Jianyu Chen. Video prediction policy: A generalist robot policy with predictive visual representations. In *International Conference on Machine Learning (ICML)*, 2025. URL https://github.com/roboterax/video-prediction-policy.

Wenlong Huang, Fei Xia, Ted Xiao, Harris Chan, Jacky Liang, Pete Florence, Andy Zeng, Jonathan Tompson, Igor Mordatch, Yevgen Chebotar, Pierre Sermanet, Tomas Jackson, Noah Brown, Linda Luu, Sergey Levine, Karol Hausman, and brian ichter. Inner Monologue: Embodied reasoning through planning with language models. In *Proceedings of The 6th Conference on Robot Learning*, volume 205 of *Proceedings of Machine Learning Research*, pp. 1769–1782. PMLR, 14–18 Dec 2023. URL https://proceedings.mlr.press/v205/huang23c.html.

Quan Huynh-Thu and Mohammed Ghanbari. Scope of validity of PSNR in image/video quality assessment. *Electronics Letters*, 44(13):800–801, 2008. doi: 10.1049/el:20080522.

Joel Jang, Seonghyeon Ye, Zongyu Lin, Jiannan Xiang, Johan Bjorck, Yu Fang, Fengyuan Hu, Spencer Huang, Kaushil Kundalia, Yen-Chen Lin, Loic Magne, Ajay Mandlekar, Avnish Narayan, You Liang Tan, Guanzhi Wang, Jing Wang, Qi Wang, Yinzhen Xu, Xiaohui Zeng, Kaiyuan Zheng, Ruijie Zheng, Ming-Yu Liu, Luke Zettlemoyer, Dieter Fox, Jan Kautz, Scott Reed, Yuke Zhu, and Linxi Fan. DreamGen: Unlocking generalization in robot learning through video world models. In *Conference on Robot Learning (CoRL)*, 2025. URL https://arxiv.org/abs/2505.12705.

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

Moo Jin Kim, Yihuai Gao, Tsung-Yi Lin, Yen-Chen Lin, Yunhao Ge, Grace Lam, Percy Liang, Shuran Song, Ming-Yu Liu, Chelsea Finn, and Jinwei Gu. Cosmos policy: Fine-tuning video models for visuomotor control and planning. *arXiv preprint arXiv:2601.16163*, 2026. URL https://arxiv.org/abs/2601.16163.

Po-Chen Ko, Jiayuan Mao, Yilun Du, Shao-Hua Sun, and Joshua B Tenenbaum. Learning to act from actionless videos through dense correspondences. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://github.com/flow-diffusion/AVDC.

Michael Laskey, Jonathan Lee, Roy Fox, Anca Dragan, and Ken Goldberg. DART: Noise injection for robust imitation learning. In *Proceedings of the 1st Annual Conference on Robot Learning*, volume 78 of *Proceedings of Machine Learning Research*, pp. 143–156. PMLR, 13–15 Nov 2017. URL https://proceedings.mlr.press/v78/laskey17a.html.

Chenxuan Li, Jiaming Liu, Guanqun Wang, Xiaoqi Li, Sixiang Chen, Liang Heng, Chuyan Xiong, Jiaxin Ge, Renrui Zhang, Kaichen Zhou, and Shanghang Zhang. A self-correcting vision-language-action model for fast and slow system manipulation. *arXiv preprint arXiv:2405.17418*, 2024. URL https://arxiv.org/abs/2405.17418.

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

Qi Li, Xingyi Yang, and Xinchao Wang. BadWAM: When world-action models dream right but act wrong. *arXiv preprint arXiv:2607.15207*, 2026b.

Shuang Li, Yihuai Gao, Dorsa Sadigh, and Shuran Song. Unified video action model. In *Robotics: Science and Systems (RSS)*, 2025.

Junbang Liang, Pavel Tokmakov, Ruoshi Liu, Sruthi Sudhakar, Paarth Shah, Rares Ambrus, and Carl Vondrick. Video generators are robot policies. *arXiv preprint arXiv:2508.00795*, 2025.

Yue Liao, Pengfei Zhou, Siyuan Huang, Donglin Yang, Shengcong Chen, Yuxin Jiang, Yue Hu, Jingbin Cai, Si Liu, Jianlan Luo, Liliang Chen, Shuicheng Yan, Maoqing Yao, and Guanghui Ren. Genie envisioner: A unified world foundation platform for robotic manipulation. In *International Conference on Learning Representations (ICLR)*, 2026. URL https://arxiv.org/abs/2508.05635.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *International Conference on Learning Representations (ICLR)*, 2023.

Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. LIBERO: Benchmarking knowledge transfer for lifelong robot learning. In *Advances in Neural Information Processing Systems*, volume 36, pp. 44776–44791. Curran Associates, Inc., 2023a. doi: 10.52202/075280-1939. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/8c3c666820ea055a77726d66fc7d447f-Paper-Datasets_and_Benchmarks.pdf.

Huihan Liu, Alice Chen, Yuke Zhu, Adith Swaminathan, Andrey Kolobov, and Ching-An Cheng. Interactive robot learning from verbal correction. *arXiv preprint arXiv:2310.17555*, 2023b. URL https://arxiv.org/abs/2310.17555.

Huihan Liu, Soroush Nasiriany, Lance Zhang, Zhiyao Bao, and Yuke Zhu. Robot learning on the job: Human-in-the-loop autonomy and learning during deployment. In *Robotics: Science and Systems (RSS)*, 2023c. URL https://github.com/ut-austin-rpl/sirius.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *International Conference on Learning Representations (ICLR)*, 2023d. URL https://github.com/gnobitab/RectifiedFlow.

Yuejiang Liu, Fan Feng, Lingjing Kong, Weifeng Lu, Jinzhou Tang, Kun Zhang, Kevin Murphy, Chelsea Finn, and Yilun Du. World Action Verifier: Self-improving world models via forward-inverse asymmetry. *arXiv preprint arXiv:2604.01985*, 2026a. URL https://arxiv.org/abs/2604.01985.

Yushan Liu, Peibo Sun, Xintao Chao, Zhenyang Yang, Yifan Xie, Lingfeng Zhang, Shoujie Li, Chenyu Tang, Fang Chen, Xiao-Ping Zhang, and Wenbo Ding. CheckVLA: Execution-time verification with action-conditioned world model for long-horizon mobile manipulation. *arXiv preprint arXiv:2607.26789*, 2026b.

Yushan Liu, Peibo Sun, Shoujie Li, Yifan Xie, Lingfeng Zhang, Xintao Chao, Shiyuan Dong, Fang Chen, Xiao-Ping Zhang, and Wenbo Ding. OA-WAM: Object-addressable world action model for robust robot manipulation. *arXiv preprint arXiv:2605.06481*, 2026c.

Zeyi Liu, Arpit Bahety, and Shuran Song. REFLECT: Summarizing robot experiences for failure explanation and correction. In *Conference on Robot Learning (CoRL)*, 2023e. URL https://github.com/real-stanford/reflect.

Weifeng Lu, Minghao Ye, Zewei Ye, Ruihan Tao, Shuo Yang, and Bo Zhao. RoboFAC: A comprehensive framework for robotic failure analysis and correction. *arXiv preprint arXiv:2505.12224*, 2025. URL https://arxiv.org/abs/2505.12224.

Yecheng Jason Ma, Joey Hejna, Ayzaan Wahid, Chuyuan Fu, Dhruv Shah, Jacky Liang, Zhuo Xu, Sean Kirmani, Peng Xu, Danny Driess, Ted Xiao, Jonathan Tompson, Osbert Bastani, Dinesh Jayaraman, Wenhao Yu, Tingnan Zhang, Dorsa Sadigh, and Fei Xia. Vision language models are in-context value learners. In *International Conference on Learning Representations (ICLR)*, 2025. URL https://arxiv.org/abs/2411.04549.

Junnan Nie, Jiayi Li, Jiachen Zhang, Junyi Lao, Chenghao Liu, Tianle Zhang, Liang Lin, and Songfang Huang. PACE: Phase-aware chunk execution for robot policies with action chunking. *arXiv preprint arXiv:2606.00537*, 2026.

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. *Transactions on Machine Learning Research*, 2024. URL https://github.com/facebookresearch/dinov2.

Yi Pan, Miao Pan, Qi Lu, Jiaming Huang, Man Zhang, Siteng Huang, Xin Li, Jie Zhang, Yongliang Shen, Xuhong Zhang, and Wenqi Zhang. VLA-Corrector: Lightweight detect-and-correct inference for adaptive action horizon. *arXiv preprint arXiv:2607.01804*, 2026.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023. URL https://github.com/facebookresearch/DiT.

Lu Qiu, Yizhuo Li, Yi Chen, Yuying Ge, Yixiao Ge, and Xihui Liu. Making Foresight Actionable: Repurposing Representation Alignment in World Action Models. *arXiv preprint arXiv:2606.12217*, 2026. doi: 10.48550/arXiv.2606.12217. URL https://arxiv.org/abs/2606.12217.

Alfredo Reichlin, Giovanni Luca Marchetti, Hang Yin, Ali Ghadirzadeh, and Danica Kragic. Back to the manifold: Recovering from out-of-distribution states. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2022. doi: 10.1109/IROS47612.2022.9981315. URL https://arxiv.org/abs/2207.08673.

Allen Z. Ren, Anushri Dixit, Alexandra Bodrova, Sumeet Singh, Stephen Tu, Noah Brown, Peng Xu, Leila Takayama, Fei Xia, Jake Varley, Zhenjia Xu, Dorsa Sadigh, Andy Zeng, and Anirudha Majumdar. Robots that ask for help: Uncertainty alignment for large language model planners. In *Conference on Robot Learning (CoRL)*, 2023. URL https://robot-help.github.io/.

Stephane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, volume 15 of *Proceedings of Machine Learning Research*, pp. 627–635, Fort Lauderdale, FL, USA, 11–13 Apr 2011. PMLR. URL https://proceedings.mlr.press/v15/ross11a.html.

Yichao Shen, Fangyun Wei, Zhiying Du, Yaobo Liang, Yan Lu, Jiaolong Yang, Nanning Zheng, and Baining Guo. VideoVLA: Video Generators Can Be Generalizable Robot Manipulators. In *Advances in Neural Information Processing Systems*, 2025. doi: 10.48550/arXiv.2512.06963. URL https://openreview.net/forum?id=UPHlqbZFZB.

Lucy Xiaoyang Shi, Zheyuan Hu, Tony Z. Zhao, Archit Sharma, Karl Pertsch, Jianlan Luo, Sergey Levine, and Chelsea Finn. Yell at your robot: Improving on-the-fly from language corrections. In *Robotics: Science and Systems (RSS)*, 2024. URL https://github.com/yay-robot/yay_robot.

Suyeon Shin, Juwon Kim, Hyeonbin Park, Hyunseo Kim, Hyundo Lee, Hyung-Sin Kim, and Byoung-Tak Zhang. Back to the Familiar Future: Failure Recovery for VLA Policies via Pre-Imagined Milestone Selection. *arXiv preprint arXiv:2606.09258*, 2026. doi: 10.48550/arXiv.2606.09258. URL https://arxiv.org/abs/2606.09258.

Rohan Sinha, Amine Elhafsi, Christopher Agia, Matt Foutter, Edward Schmerling, and Marco Pavone. Real-time anomaly detection and reactive planning with large language models. In *Proceedings of Robotics: Science and Systems*, Delft, Netherlands, July 2024. doi: 10.15607/RSS.2024.XX.114. URL https://www.roboticsproceedings.org/rss20/p114.html.

Daniel Sliwowski and Dongheui Lee. ConditionNET: Learning preconditions and effects for execution monitoring. *IEEE Robotics and Automation Letters*, 10(2):1337–1344, 2025. doi: 10.1109/LRA.2024.3520916. URL https://dsliwowski1.github.io/ConditionNET_page/.

Sruthi Sudhakar, Junbang Liang, Sreehari Rammohan, Pavel Tokmakov, Richard Zemel, and Carl Vondrick. Robot critics that sweat the small stuff. *arXiv preprint arXiv:2606.21572*, 2026.

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, Jiayu Wang, Jingfeng Zhang, Jingren Zhou, Jinkai Wang, Jixuan Chen, Kai Zhu, Kang Zhao, Keyu Yan, Lianghua Huang, Mengyang Feng, Ningyi Zhang, Pandeng Li, Pingyu Wu, Ruihang Chu, Ruili Feng, Shiwei Zhang, Siyang Sun, Tao Fang, Tianxing Wang, Tianyi Gui, Tingyu Weng, Tong Shen, Wei Lin, Wei Wang, Wei Wang, Wenmeng Zhou, Wente Wang, Wenting Shen, Wenyuan Yu, Xianzhong Shi, Xiaoming Huang, Xin Xu, Yan Kou, Yangyu Lv, Yifei Li, Yijing Liu, Yiming Wang, Yingya Zhang, Yitong Huang, Yong Li, You Wu, Yu Liu, Yulin Pan, Yun Zheng, Yuntao Hong, Yupeng Shi, Yutong Feng, Zeyinzi Jiang, Zhen Han, Zhi-Fan Wu, and Ziyu Liu. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025. URL https://arxiv.org/abs/2503.20314.

Santosh Thoduka, Sebastian Houben, Juergen Gall, and Paul G. Plöger. Enhancing video-based robot failure detection using task knowledge. In *European Conference on Mobile Robots (ECMR)*, pp. 1–6. IEEE, 2025. doi: 10.1109/ECMR65884.2025.11162998.

Yang Tian, Sizhe Yang, Jia Zeng, Ping Wang, Dahua Lin, Hao Dong, and Jiangmiao Pang. Predictive inverse dynamics models are scalable learners for robotic manipulation. In *International Conference on Learning Representations (ICLR)*, 2025. URL https://arxiv.org/abs/2412.15109.

Haoxuan Wang, Gengyu Zhang, Yan Yan, Ramana Rao Kompella, and Gaowen Liu. VLA knows its limits: Adaptive execution horizons for robot policies. *arXiv preprint arXiv:2602.21445*, 2026a.

Rui Wang, Yue Zhang, Jiehong Lin, Kuncheng Luo, Jianan Wang, Zhongrui Wang, and Xiaojuan Qi. When to trust imagination: Adaptive action execution for world action models. *arXiv preprint arXiv:2605.06222*, 2026b.

Hongtao Wu, Ya Jing, Chilam Cheang, Guangzeng Chen, Jiafeng Xu, Xinghang Li, Minghuan Liu, Hang Li, and Tao Kong. Unleashing large-scale video generative pre-training for visual robot manipulation. In *International Conference on Learning Representations*, 2024a. URL https://github.com/bytedance/GR-1.

Jialong Wu, Shaofeng Yin, Ningya Feng, Xu He, Dong Li, Jianye Hao, and Mingsheng Long. iVideoGPT: Interactive videogpts are scalable world models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024b. URL https://github.com/thuml/iVideoGPT.

Chen Xu, Tony Khuong Nguyen, Emma Dixon, Christopher Rodriguez, Patrick Miller, Robert Lee, Paarth Shah, Rares Andrei Ambrus, Haruki Nishimura, and Masha Itkina. Can we detect failures without failure data? uncertainty-aware runtime failure detection for imitation learning policies. In *Proceedings of Robotics: Science and Systems*, 2025. doi: 10.15607/RSS.2025.XXI.073. URL https://www.roboticsproceedings.org/rss21/p073.html.

Sherry Yang, Yilun Du, Kamyar Ghasemipour, Jonathan Tompson, Leslie Kaelbling, Dale Schuurmans, and Pieter Abbeel. Learning interactive real-world simulators. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://arxiv.org/abs/2310.06114.

Angen Ye, Boyuan Wang, Chaojun Ni, Guan Huang, Guosheng Zhao, Hao Li, Hengtao Li, Jie Li, Jindi Lv, Jingyu Liu, Min Cao, Peng Li, Qiuping Deng, Wenjun Mei, Xiaofeng Wang, Xinze Chen, Xinyu Zhou, Yang Wang, Yifan Chang, Yifan Li, Yukun Zhou, Yun Ye, Zhichao Liu, and Zheng Zhu. GigaWorld-Policy: An efficient action-centered world–action model. *arXiv preprint arXiv:2603.17240*, 2026a. URL https://arxiv.org/abs/2603.17240.

Seonghyeon Ye, Joel Jang, Byeongguk Jeon, Sejune Joo, Jianwei Yang, Baolin Peng, Ajay Mandlekar, Reuben Tan, Yu-Wei Chao, Bill Yuchen Lin, Lars Liden, Kimin Lee, Jianfeng Gao, Luke Zettlemoyer, Dieter Fox, and Minjoon Seo. Latent action pretraining from videos. In *International Conference on Learning Representations (ICLR)*, 2025. URL https://arxiv.org/abs/2410.11758.

Seonghyeon Ye, Yunhao Ge, Kaiyuan Zheng, Shenyuan Gao, Sihyun Yu, George Kurian, Suneel Indupuru, You Liang Tan, Chuning Zhu, Jiannan Xiang, Ayaan Malik, Kyungmin Lee, William Liang, Nadun Ranawaka, Jiasheng Gu, Yinzhen Xu, Guanzhi Wang, Fengyuan Hu, Avnish Narayan, Johan Bjorck, Jing Wang, Gwanghyun Kim, Dantong Niu, Ruijie Zheng, Yuqi Xie, Jimmy Wu, Qi Wang, Ryan Julian, Danfei Xu, Yilun Du, Yevgen Chebotar, Scott Reed, Jan Kautz, Yuke Zhu, Linxi "Jim" Fan, and Joel Jang. World action models are zero-shot policies. *arXiv preprint arXiv:2602.15922*, 2026b. URL https://arxiv.org/abs/2602.15922.

Tianyuan Yuan, Zibin Dong, Yicheng Liu, and Hang Zhao. Fast-WAM: Do world action models need test-time future imagination? *arXiv preprint arXiv:2603.16666*, 2026. doi: 10.48550/arXiv.2603.16666. URL https://arxiv.org/abs/2603.16666.

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*, 2026a.

Xiaoxiong Zhang, Xiong Zeng, and Wei Zhang. From world models to world action models: A concise tutorial for robotics. *arXiv preprint arXiv:2607.00836*, 2026b. URL https://arxiv.org/abs/2607.00836.

Yanyan Zhang, Disheng Liu, Kai Ye, Chaoda Song, Xinpeng Li, Mohsen Hariri, Vikash Singh, Yu Yin, and Vipin Chaudhary. Imagining Recovery: Inference-Time Counterfactual Realignment for Vision-Language-Action Models. *arXiv preprint arXiv:2608.14822*, 2026c. doi: 10.48550/arXiv.2608.14822. URL https://arxiv.org/abs/2608.14822.

Guiyu Zhao, Longteng Guo, Junyou Zhu, Jun Fu, Yanghong Mei, Bin Cao, Jie Jiang, Xingjian He, and Jing Liu. VeriSpace: Spatially grounded action verification for vision-language-action models. *arXiv preprint arXiv:2606.10568*, 2026.

Enshen Zhou, Qi Su, Cheng Chi, Zhizheng Zhang, Zhongyuan Wang, Tiejun Huang, Lu Sheng, and He Wang. Code-as-Monitor: Constraint-aware visual programming for reactive and proactive robotic failure detection. *arXiv preprint arXiv:2412.04455*, 2024a.

Pengfei Zhou, Shengcong Chen, Di Chen, Jiaxu Wang, Rongjun Jin, Bingwen Zhu, Yike Pan, Songen Gu, Kuanning Wang, Shufeng Nan, Xingyu Qiu, Chenhao Qiu, Pu Yang, Yunuo Cai, Jianxiong Gao, Yifan Li, Yanwei Fu, Xiangyu Yue, Zhi Chen, and Jianlan Luo. τ_0-WM: A unified video-action world model for robotic manipulation. *arXiv preprint arXiv:2606.01027*, 2026. URL https://arxiv.org/abs/2606.01027.

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, 2024b. URL https://github.com/rainbow979/robodreamer.

Chuning Zhu, Raymond Yu, Siyuan Feng, Benjamin Burchfiel, Paarth Shah, and Abhishek Gupta. Unified world models: Coupling video and action diffusion for pretraining on large robotic datasets. In *Proceedings of Robotics: Science and Systems*, Los Angeles, CA, USA, June 2025a. doi: 10.15607/RSS.2025.XXI.015. URL https://github.com/WEIRDLabUW/unified-world-model.

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 (ICCV)*, pp. 9834–9844, October 2025b. URL https://openaccess.thecvf.com/content/ICCV2025/html/Zhu_IRASim_A_Fine-Grained_World_Model_for_Robot_Manipulation_ICCV_2025_paper.html.