BADWAM: WHEN WORLD-ACTION MODELS DREAM RIGHT BUT ACT WRONG

Anonymous authors
Paper under double-blind review

ABSTRACT

World-action models (WAMs) are emerging as a promising foundation for embodied control: rather than predicting actions alone, they learn representations that couple action generation with future world prediction. This coupling is often viewed as a source of robustness, interpretability, and safety, as a robot's action can in principle be checked against its imagined future. In this paper, we show that this assumption is fragile. We introduce *BadWAM*, a unified framework for modeling *World-Action Drift Attacks*: a new class of WAM-specific adversarial attacks that use small visual perturbations to break the alignment between what a WAM imagines and what it executes. BadWAM characterizes this attack surface along two natural criteria: attack strength and stealthiness. When the adversary prioritizes disruption, BadWAM instantiates an *action-only adversarial attack*, which directly drives the model toward task-failing actions. When the adversary additionally prioritizes stealth, BadWAM instantiates an *imagination-preserving adversarial attack*, which seeks to induce harmful action shifts while keeping the model's predicted future close to its clean imagination. Together, these two attacks capture a spectrum of WAM-specific failures: from overt action hijacking to stealthier cases where the model appears to imagine a plausible future but executes a desynchronized action. We evaluate BadWAM across multiple WAM variants and benchmarks. Results show that our attacks substantially reduce task success rates under different execution setups. For example, our action-only attack reduces the model performance from 96.5% to 43.1% success on LIBERO. The results of our imagination-preserving attack further exposes a WAM-specific vulnerability: moderate future-preserving regularization can maintain strong attack performance while reducing future imagination drift.

1 INTRODUCTION

Robotic foundation models are rapidly moving from action prediction toward *world-action modeling* (Kim et al., 2025; Black et al., 2024; Intelligence et al., 2026; Zitkovich et al., 2023; Yuan et al., 2026; Shen et al., 2026b). A world-action model (WAM) does not merely map observations and language instructions to low-level controls; it also learns representations that couple action generation with future world prediction (Yuan et al., 2026; Shen et al., 2026b). This design is attractive for embodied AI. Future prediction can provide richer physical representations, enable planning or verification, and offer an intuitive safety signal: if a robot can imagine the consequences of its actions, then unsafe or inconsistent behavior might be detected before execution (Shen et al., 2026b; Yuan et al., 2026; Li et al., 2026).

Recent WAMs instantiate this idea in different ways. Some models explicitly generate future observations or latent futures before action prediction, while others use video modeling as an auxiliary training objective and skip explicit future generation at test time (Yuan et al., 2026). Newer WAM variants further extend this paradigm toward object-addressable manipulation, mobile manipulation, and visual-tactile contact-rich control (Liu et al., 2026b; Chen et al., 2026b; Tian et al., 2026). Across these designs, the central promise is the same: coupling action with world prediction should make robot behavior more physically grounded. This promise also motivates emerging WAM safety mechanisms that inspect imagined futures, latent rollouts, or visual trajectories to detect unsafe plans (Liu et al., 2026a; Chen et al., 2026a).

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.

This paper begins with a basic security question: *are WAMs robust to adversarially induced execution failures?* We find that WAM coupling does not guarantee robustness: bounded visual perturbations can reduce task success across WAM variants. More specifically, *action generation and future prediction can fail asynchronously*: the action output can shift substantially even when the predicted future remains close to its clean counterpart. This failure mode is qualitatively different from standard adversarial attacks on image classifiers, visual world models, or reactive robot policies (Szegedy et al., 2014; Goodfellow et al., 2015; Madry et al., 2018; Shen et al., 2026a). The attacker is not merely changing a label or shifting an action vector; it is exploiting the interface between two coupled capabilities: imagining and acting.

We call this vulnerability *World-Action Drift Attack*. Rather than treating action and imagination as independent outputs, BadWAM focuses on their alignment. A bounded visual perturbation can shift the model toward task-failing behavior while leaving the predicted future comparatively close to its clean prediction. This creates a dangerous gap: a robot may appear to imagine a reasonable future, yet execute an action that is no longer aligned with that imagination. In such cases, future prediction alone is not a reliable safety signal.

We instantiate this idea in *BadWAM*, a unified framework for modeling and evaluating world-action drift attacks under black-box access to a deployed WAM. BadWAM characterizes the attack surface along two natural criteria: attack strength and stealthiness. When the adversary prioritizes disruption, BadWAM instantiates an *action-only adversarial attack*, which drives the model toward task-failing actions using only queryable model outputs. This attack captures the overt end of the threat spectrum, where the adversary aims to maximize execution failure. When the adversary additionally prioritizes stealth, BadWAM instantiates an *imagination-preserving adversarial attack*. This attack also operates through black-box queries, but optimizes for harmful action shifts while keeping the model's predicted future close to its clean imagination. Intuitively, this creates a stealthier failure: the robot appears to imagine a plausible future, but the action it executes has been desynchronized from that future. Together, they expose a spectrum of black-box WAM failures, from overt action hijacking to imagination-preserving failure.

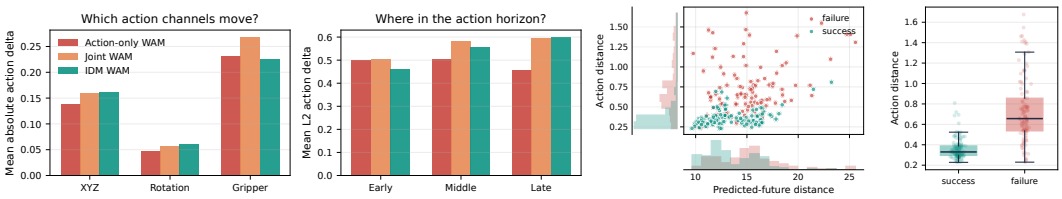
We evaluate BadWAM on LIBERO (Liu et al., 2023) and RoboTwin (Chen et al., 2025). We do not conduct real world experiments because the untargeted nature of our attacks may cause irreversible damage to certain entities or the robot itself. Our experiments show that WAMs are vulnerable across variants. The action-only attack substantially reduces closed-loop task success, and the imagination-preserving attack exposes a distinct WAM-specific risk: moderate future-preserving regularization can maintain strong attack performance while reducing imagination drift. In addition, BadWAM transfers effectively across WAM variants, performing very close to the attacks optimized directly on the target. This is security-critical: the vulnerability generalizes beyond a specific action head, allowing an attacker to compromise related WAM variants without target-specific optimization.

This work makes the following contributions:

- We identify *World-Action Drift Attack* as a WAM-specific attack surface, where action outputs can be adversarially hijacked while the model's predicted futures remain comparatively close to their clean counterparts.
- We design *BadWAM*, a unified framework with two attack instantiations: *action-only adversarial attacks* for maximizing execution disruption, and *imagination-preserving adversarial attacks* for causing failures while preserving plausible predicted futures.
- We evaluate BadWAM on different robot manipulation tasks, showing substantial task degradation under bounded visual perturbations. We also provide detailed analyses of attack dynamics, offering insight into why WAMs fail under world-action drift attacks.

2 Related Work

World Action Models. World-action models have recently emerged as a predictive-action paradigm for embodied AI (Shen et al., 2026b). Unlike conventional visuomotor policies that directly map observations and instructions to actions, WAMs expose future prediction as part of the action-generation process. Let o_t denote the current visual observation, g denote a language instruction or goal, and $a_{t:t+H-1}$ denote an action chunk over horizon H. A conventional reactive policy can



(a) Structured action shifts under action-only attacks.

(b) Desynchronization evidence.

Figure 1: **Empirical motivation for world-action adversarial attacks.** **(a)** Action-only attacks produce structured shifts in continuous action channels and specific portions of the action horizon, rather than uniform output noise. **(b)** Failed episodes tend to exhibit larger action shifts, whereas predicted-future shifts substantially overlap between successful and failed executions, motivating attacks on the alignment between action and imagination.

be abstracted as

$$a_{t:t+H-1} \sim \pi_\theta(o_t, g). \tag{1}$$

By contrast, a WAM additionally models future world states, such as latent future states $z_{t+1:t+K}$ or decoded future frames $v_{t+1:t+K}$. Abstractly, its world-prediction module W_θ first imagines a future, and its action module A_θ then produces an action sequence conditioned on that imagination:

$$z_{t+1:t+K} \sim W_\theta(o_t, g), \quad a_{t:t+H-1} \sim A_\theta(o_t, g, z_{t+1:t+K}). \tag{2}$$

In fully joint formulations, the model may instead learn a joint distribution over future states and actions:

$$(z_{t+1:t+K}, a_{t:t+H-1}) \sim p_\theta(z_{t+1:t+K}, a_{t:t+H-1} \mid o_t, g). \tag{3}$$

In all cases, the defining feature is that action prediction is coupled, explicitly or implicitly, with a learned model of how the world may evolve.

Attacks against Embodied AI Systems. Adversarial attacks are first studied in image classification, where small perturbations can alter model predictions (Szegedy et al., 2014; Goodfellow et al., 2015; Madry et al., 2018). Later work extended them to black-box settings (Chen et al., 2017; Ilyas et al., 2018; Uesato et al., 2018) and show their effectiveness under physical transformations (Brown et al., 2017; Eykholt et al., 2018). In embodied systems, adversarial observations and environmental cues can manipulate reinforcement-learning agents and control policies (Lin et al., 2017; Gleave et al., 2020). Recent work reveals vulnerabilities in predictive embodied models. BadWorld degrades visual-world-model rollouts through context-image perturbations Shen et al. (2026a); physical-conditioned attacks manipulate conditioning signals to induce semantic or decision errors Guo et al. (2026); and JailWAM elicits unsafe visual trajectories from WAM controllers Liu et al. (2026a). These studies target either policy behavior or predicted futures.

The closest work studies integrity attacks against imagine-then-act systems whose predicted futures are consumed by safety gates, planners, or verifiers Chen et al. (2026a). BadWAM instead targets the consistency between imagination and action. It shows that an attacker can substantially alter the executed action while keeping the imagined future close to its clean prediction, potentially bypassing monitors that trust plausible rollouts. We study this failure mode through an action-only attack that prioritizes task disruption and an imagination-preserving attack that additionally optimizes stealth.

3 Threat Model

We study inference-time attacks against deployed WAM-based robot policies. Following the notation introduced above, the robot observes o_t and receives an instruction or goal g at each replanning step. The WAM then outputs an action chunk $a_{t:t+H-1}$ and, depending on the model interface, may also expose an imagined future in latent form $z_{t+1:t+K}$ or decoded video form $v_{t+1:t+K}$. The robot executes part of the predicted action chunk, observes the environment again, and repeats this process in closed loop.

Adversarial capability. The adversary can perturb the visual observation before it is processed by the WAM. We model a digital observation-channel adversary that can intercept the current camera tensor, submit black-box queries, and replace the tensor delivered to the policy. We do not claim

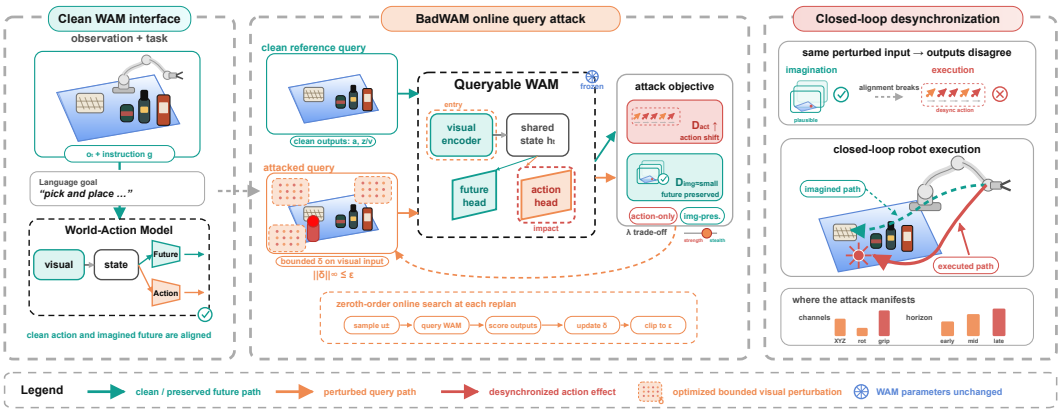


Figure 2: **Overview of BadWAM.** BadWAM injects a small visual perturbation into model observations and performs query-based online search over a frozen WAM. The optimized perturbation disrupts the action prediction pathway while preserving or minimally altering visual rollout predictions, leading to world-action adversarial attacks during closed-loop execution.

that the same perturbations are directly realizable as physical-world patterns. We write the perturbed observation as

$$\tilde{o}_t = \mathrm{clip}(o_t + \delta_t), \quad \|\delta_t\|_\infty \le \epsilon. \tag{4}$$

The perturbation is bounded so that the attacked observation remains visually close to the clean observation. For multi-camera inputs, we treat o_t as the concatenation of all camera observations and apply the same norm constraint to the full visual input. The adversary alters only the visual observation, leaving the instruction, robot state, environment dynamics and model parameters unchanged.

We consider two access levels. In the *action-only* setting, the adversary can query the WAM and observe only its output action chunk. This captures black-box access to a deployed policy API or robot inference stack. In the *imagination-visible* setting, the adversary can additionally observe the model's imagined future, either as latent rollouts or decoded future video. This captures systems where future prediction is exposed for planning, debugging, monitoring, or imagine-then-check safety verification (Chen et al., 2026a; Zhao et al., 2024). In both settings, the adversary does not require gradients or white-box access to model weights.

Adversarial objective. The adversary's primary goal is untargeted task failure: the robot should fail to complete the instructed task under closed-loop execution. The attack is successful if a perturbation causes the robot to execute a sequence of actions that reduces task success, even though the instruction and environment remain unchanged. Since task reward and success labels are unavailable during optimization, BadWAM uses clean-attacked action deviation as a queryable surrogate objective. Besides attack strength, BadWAM additionally studies stealthiness. A non-stealthy attack may simply maximize action disruption. A stealthier attack should degrade execution while keeping the WAM's imagined future close to its clean prediction. This is important because future prediction is often treated as a safety signal: a monitor may inspect the imagined future and accept the action if the future appears plausible (Chen et al., 2026a). BadWAM targets precisely this gap between what the WAM imagines and what it executes.

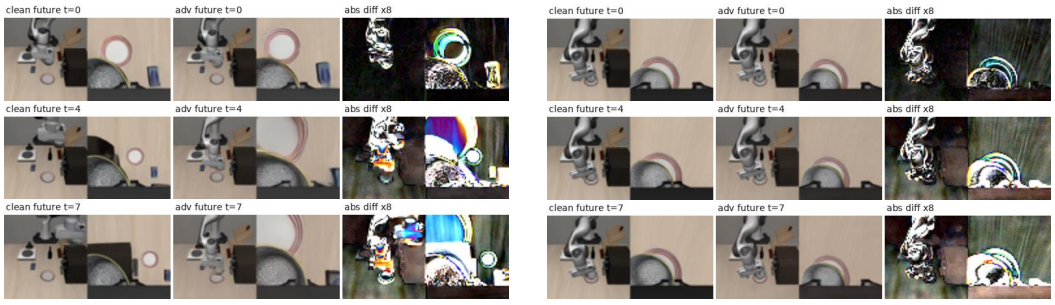
4 BadWAM

BadWAM is a unified framework for modeling and evaluating world-action drift attacks. As shown in Figure 2, the framework treats a WAM as a queryable predictive-action system and optimizes bounded visual perturbations at each replanning step. Its two attack instantiations correspond to two points on the attack strength and stealthiness: an *action-only adversarial attack* that prioritizes execution failure, and an *imagination-preserving adversarial attack* that additionally constrains predicted-future drift.

Design Motivation. BadWAM is motivated by a simple observation: action and imagination are coupled in WAMs, but they are not inseparable. In our closed-loop traces, task failures tend to exhibit

Table 1: **Main attack results.** Results use $\epsilon = 0.06$ and eight optimization iterations ($B = 16$ perturbation queries). Gray values show the absolute reduction from clean task success.

	LIBERO (%)			RoboTwin (%)		
Model	Clean	Action-only	Img-pres.	Clean	Action-only	Img-pres.
Action-only WAM	96.5	43.1 (↓53.4)	–	92.1	84.4 (↓7.7)	–
Joint WAM	98.1	61.5 (↓36.6)	63.0 (↓35.1)	93.3	86.0 (↓7.3)	87.0 (↓6.3)
IDM WAM	98.4	66.1 (↓32.3)	67.0 (↓31.4)	92.9	75.4 (↓17.5)	76.5 (↓16.4)



(a) Imagined future without preservation.

(b) Imagined future with preservation.

Figure 4: **Qualitative illustration of the imagination-preserving objective on IDM WAM.** Clean and adversarial predicted futures are compared at selected steps; `abs diff` shows their $8\times$-amplified pixel difference. Both attacks induce action failure, while the preservation term keeps the adversarial future closer to the clean prediction.

larger action shifts than successful episodes, while the corresponding imagination shifts can substantially overlap. This suggests that the security-critical object is not only the action or the imagination alone, but their alignment. Figure 1b visualizes this phenomenon under our imagination-preserving attack. This motivation is also consistent with recent diagnostic evidence that WAM action heads can rely on future-predictive latents in a task-phase-dependent manner (Mishra et al., 2026). BadWAM asks a complementary security question: whether this coupling can be adversarially utilized.

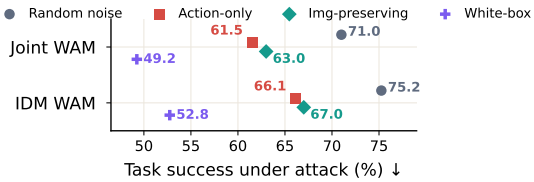


Figure 3: **Comparison with random perturbations on LIBERO.** All methods use the same ℓ_∞ budget. Lower value indicates a stronger attack.

We therefore formulate BadWAM around two distances. D_{act} measures the deviation between clean and attacked action chunks, and D_{img} measures the drift between clean and attacked imagination. These distances are interface-level quantities: they do not require a particular WAM architecture, only access to the corresponding outputs. Conceptually, a stealthy adversarial attack seeks large action deviation under bounded visual perturbations while keeping imagination drift below a monitorable threshold:

$$\max_{\|\delta\|_\infty \le \epsilon} D_{\text{act}}(a^\delta, a) \quad \text{s.t. } D_{\text{img}}(z^\delta, z) \le \tau_{\text{img}}. \tag{5}$$

Equation 5 defines the core BadWAM attack surface. The two attacks below instantiate different relaxations of this objective.

Action-Only Adversarial Attack. The action-only attack captures the high-strength end of BadWAM. It treats the WAM as a queryable controller and focuses purely on execution disruption. Rather than enforcing any constraint on the imagined future, it searches for a bounded visual perturbation that maximizes the deviation between clean and attacked action chunks:

$$\delta_t^\star = \arg\max_{\|\delta_t\|_\infty \le \epsilon} D_{\text{act}}\left(a^\delta_{t:t+H-1}, a_{t:t+H-1}\right). \tag{6}$$

The clean action chunk is used only as a reference; the attacker does not need ground-truth expert actions or task success labels during optimization. This makes the attack applicable to deployed policy interfaces where the observable output is the action command sent to the robot controller.

Although simple, this objective is already highly disruptive. In our closed-loop experiments, the action-only attack reduces one WAM variant from 96.5% to 43.1% task success. We therefore use

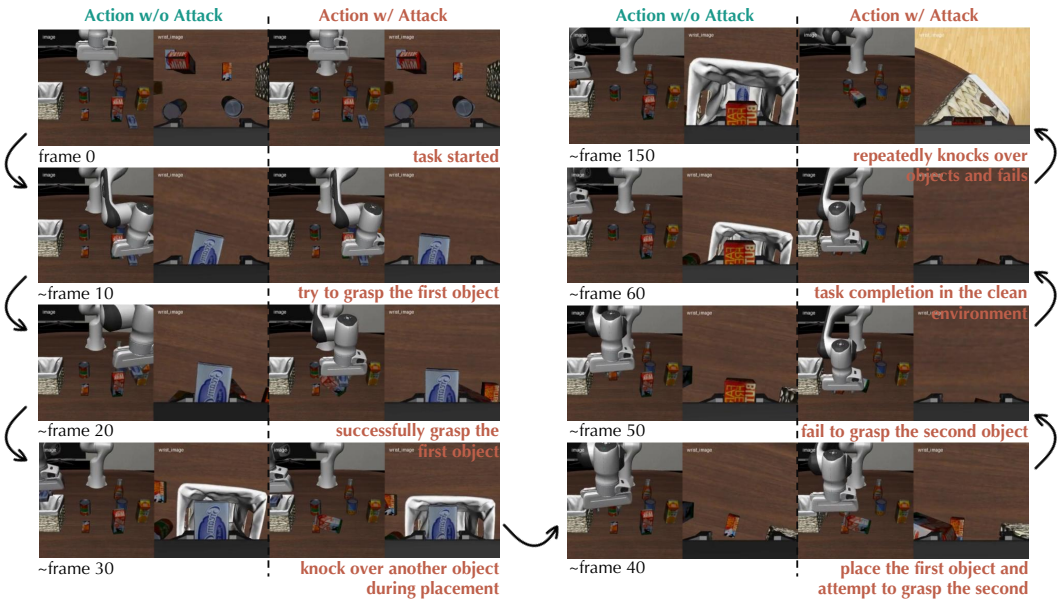


Figure 5: **Qualitative comparison of clean and attacked rollouts on a LIBERO-10 task.** Each row shows synchronized third-person and wrist-camera observations over time. The clean WAM completes the sequential manipulation task, while the attacked WAM initially behaves plausibly but gradually drifts, knocks over objects, fails to grasp the second object, and eventually fails.

it as the high-strength endpoint of BadWAM. At the same time, the attack is not merely unstructured output corruption: the induced shifts are unevenly distributed across both action channels and temporal horizon segments. For example, Figure 1a shows that the perturbations concentrate on task-relevant channels such as translation and gripper commands, while their temporal concentration varies across WAM variants. This motivates logging channel-level and horizon-level action statistics in our evaluation.

Imagination-Preserving Adversarial Attack. The action-only objective provides no control over what happens to the imagined future. For WAM deployments that expose imagined futures to planners, monitors, or imagine-then-check safety gates, this is insufficient: a stealthier attack should degrade execution while preserving the model's clean imagination. BadWAM therefore instantiates an imagination-preserving attack by solving the Lagrangian relaxation of Equation 5:

$$\delta_t^\star = \arg\max_{\|\delta_t\|_\infty \le \epsilon} D_{\text{act}}(a^\delta_{t:t+H-1}, a_{t:t+H-1}) - \lambda D_{\text{img}}(z^\delta_{t+1:t+K}, z_{t+1:t+K}). \tag{7}$$

The coefficient λ controls the trade-off between attack-strength and stealthiness. When $\lambda = 0$, the objective reduces to pure action disruption. As λ increases, the optimizer is discouraged from changing the imagined future, making the attack more stealthy but potentially less damaging.

When decoded future videos are available, we instantiate the imagination distance as an average frame-level distance:

$$D_{\text{img}}(v^\delta_{t+1:t+K}, v_{t+1:t+K}) = \frac{1}{K}\sum_{k=1}^{K} d_{\text{frame}}(v^\delta_{t+k}, v_{t+k}). \tag{8}$$

The same objective can be applied to latent futures by replacing the frame-level distance with a latent-space distance. In either case, the attack targets a WAM-specific failure: the robot may appear to imagine a plausible future while executing a desynchronized action.

Query-Based Online Optimization. BadWAM does not backpropagate through the WAM or access model parameters. At each replanning step, the attacker treats the WAM as a queryable input-output system and optimizes the attack objective with zeroth-order finite-difference queries. Concretely, for a current perturbation δ_t, the attacker samples random directions u_i and evaluates the scalar objective under positively and negatively perturbed observations. This gives the estimator

$$\widehat{\nabla} J(\delta_t) = \frac{1}{m}\sum_{i=1}^{m} \frac{J(\delta_t + cu_i) - J(\delta_t - cu_i)}{2c} u_i, \tag{9}$$

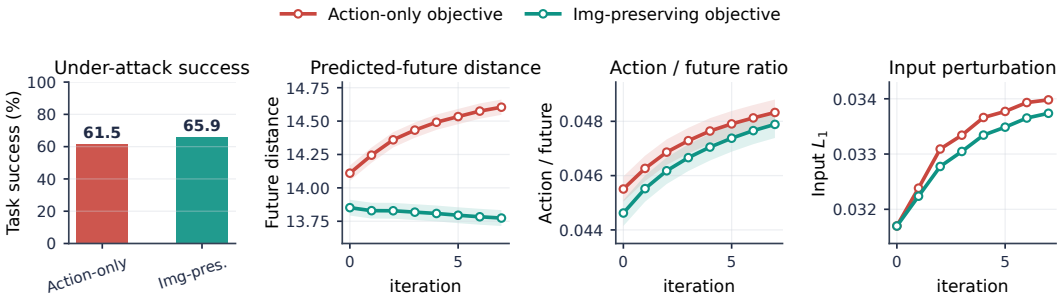


Figure 6: **Matched-strength stealth trade-off.** The imagination-preserving objective produces consistently smaller predicted-future shifts under a comparable input perturbation budget.

where J denotes either the action-only objective or the imagination-preserving objective, c is the finite-difference radius, and m is the number of sampled directions. The perturbation is then updated and projected back into the feasible ℓ_∞ ball:

$$\delta_t \leftarrow \Pi_{[-\epsilon,\epsilon]} \left(\delta_t + \eta \widehat{\nabla} J(\delta_t) \right). \tag{10}$$

This optimization is zeroth-order because it only uses objective values computed from WAM outputs; it does not require gradients, model weights, or training data. In our implementation, the random directions follow a simultaneous perturbation strategy (Spall, 1992; Uesato et al., 2018). The same optimizer supports both BadWAM instantiations by changing only the scalar objective J: the action-only attack scores action deviation, while the imagination-preserving attack scores action deviation together with future-prediction consistency.

Qualitative Example. Figure 4 qualitatively illustrates the role of the preservation term. Both variants can induce action failure, but adding preservation keeps the predicted future closer to the clean imagination. This is precisely the WAM-specific failure mode BadWAM targets: the model's imagination remains plausible while the executed action is desynchronized.

Positioning. We *do not* execute BadWAM on physical robots because the evaluation deliberately searches for task-failing controls whose consequences cannot be bounded in advance. Even small observation perturbations may accumulate into abrupt contacts, collisions, dropped objects, or motions outside the safe workspace, posing risks to hardware and nearby operators. We therefore conduct the closed-loop evaluation in simulation, which enables repeatable initial conditions, matched clean and attacked rollouts, and systematic full-sweep testing without avoidable physical harm. Real-robot validation would require an independent safety layer, conservative motion and force limits, and hardware-specific supervision, which we leave to future work.

5 Evaluation

We organize the evaluation around six research questions: **RQ1-Attack effectiveness.** How much can bounded observation perturbations reduce closed-loop WAM reliability? **RQ2-Failure mechanism.** What do BadWAM failures look like during execution, and how does the query-based search produce them? **RQ3-Stealth under comparable attack effectiveness:** At comparable attack effectiveness, what does imagination preservation buy? **RQ4-Transferability:** Do the perturbations transfer across WAM variants? **RQ6-Sensitivity and efficiency:** How do future preservation, perturbation size, and query budget trade off?

Benchmarks. Our experiments use two robot manipulation benchmarks: LIBERO Liu et al. (2023) and RoboTwin Chen et al. (2025). Unless otherwise stated, main results cover all 40 LIBERO and 50 RoboTwin tasks, with 20 trials per task. Clean and attacked conditions use the same task definitions and evaluation seeds; deviations are noted in the captions.

WAM policies. We evaluate three WAM controllers with distinct interfaces (Yuan et al., 2026; Shen et al., 2026b): an action-only WAM that directly predicts actions, a joint WAM that predicts future states and actions together, and an IDM WAM that decodes actions from future-imagination representations. The action-only model uses official checkpoints, while the joint and IDM variants are trained on 8 H100 GPUs with default settings (Yuan et al., 2026). Details are in Appendix B.1.

Attacks. We study two objectives: the *action-only adversarial attack*, which maximizes action shifts

Table 2: **Transferability of attacks across WAM variants on LIBERO.** "Direct on target" optimizes the attack directly on the target. Gray values show the reduction from target-clean success; $D_{\text{img}}^{\text{direct}}$ and $D_{\text{img}}^{\text{transfer}}$ denote mean predicted-future distances for direct and transferred attacks.

Attack objective	Source → Target	Direct on target (%)	Transferred attack (%)	$D_{\text{img}}^{\text{direct}}$	$D_{\text{img}}^{\text{transfer}}$
Action-only	Action-only WAM → Joint WAM	61.5 (−36.6%)	⟶ 64.1 (−34.0%)	–	–
Action-only	Joint WAM → IDM WAM	66.1 (−32.3%)	⟶ 70.8 (−27.6%)	–	–
Action-only	IDM WAM → Joint WAM	61.5 (−36.6%)	⟶ 65.4 (−32.7%)	–	–
Img-pres.	Joint WAM → IDM WAM	67.0 (−31.4%)	⟶ 71.3 (−27.1%)	14.18	⟶ 13.85
Img-pres.	IDM WAM → Joint WAM	63.0 (−35.1%)	⟶ 65.0 (−33.1%)	14.24	⟶ 14.16

using only action outputs, and the *imagination-preserving adversarial attack*, which additionally penalizes future drift and requires query-visible future predictions. Unless otherwise stated, both attacks perturb the full normalized image with $\epsilon = 0.06$ (approximately $8/255$ in RGB space) and use eight paired finite-difference updates per replan. This corresponds to $B = 16$ perturbation queries plus one clean reference query, or 17 total queries. Ablations are reported in Section 5.3.

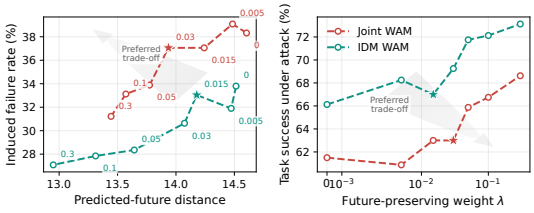


Figure 7: **Ablation on future-preserving weight λ.** Moderate future preservation can improve the action-future tradeoff, while excessive preservation weakens action manipulation.

Metrics. Our primary metric is closed-loop task success rate; lower success under attack indicates a stronger attack. For a benchmark with T tasks and N_i evaluation episodes for task i, we compute

$$\text{Success} = \frac{1}{T}\sum_{i=1}^{T}\frac{1}{N_i}\sum_{j=1}^{N_i}\mathbb{I}[s_{i,j}=1], \tag{11}$$

where $s_{i,j}$ indicates whether episode j of task i succeeds. We also report the induced success drop. For mechanism analysis, we use action distance D_{act}, predicted-future distance D_{img}, and the normalized decoupling score from Section 4.

5.1 Main Results

BadWAM Reliably Induces Task Failures. To answer **RQ1**, Table 1 summarizes the main results on LIBERO and RoboTwin. Across all WAM variants, bounded observation perturbations substantially reduce success; for example, the action-only WAM on LIBERO drops from 96.5% to 43.1%. The attacks also generalize to RoboTwin, while imagination-preserving attacks retain comparable strength with reduced predicted-future drift. These results show that coupling action generation with future prediction does not inherently protect WAMs from world-action drift attacks.

To distinguish targeted attacks from generic visual fragility, Figure 3 compares BadWAM with a model-agnostic random perturbation baseline and a stronger white-box reference under the same ℓ_∞ budget. The white-box attack optimizes the input through the WAM's action output but is still evaluated in closed loop. It lies outside BadWAM's threat model and serves as an upper-bound reference for gradient-access attacks. As can be observed, random noise is weaker than BadWAM: success remains 71.0% and 75.2% on the Joint and IDM WAMs, compared with 61.5%/63.0% and 66.1%/67.0% under action-only/imagination-preserving attacks. The white-box reference lowers success further to 49.2% and 52.8%, confirming targeted vulnerability and room for stronger gradient-access attacks.

How Do BadWAM Failures Emerge? To answer **RQ2**, we analyze representative rollouts and optimization traces. Figure 5 shows that the clean WAM completes the sequential manipulation task by grasping and placing the required objects in order. The attacked WAM, however, initially behaves plausibly but progressively drifts from the task objective, eventually knocking over objects and failing to grasp the second one. Dataset-level drift and perturbation spatial structure are analyzed in Appendices B.6.3 and B.6.4, respectively.

Imagination Preservation at Comparable Attack Effectiveness. To answer **RQ3**, we compare both objectives under matched perturbation bounds, query budgets, and closed-loop protocols with comparable task effectiveness. Figure 6 shows similar attack success, while imagination preserva-

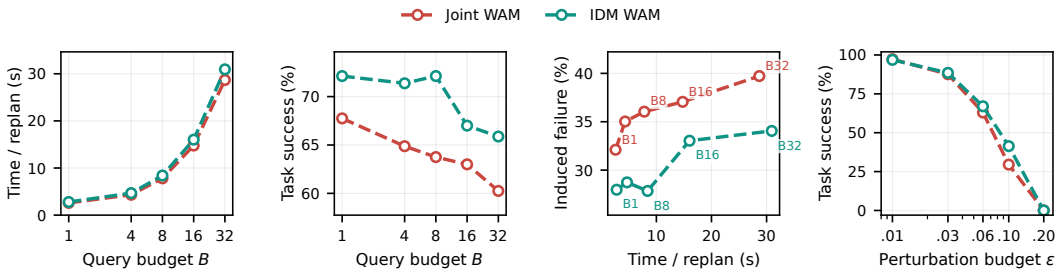


Figure 8: **Efficiency and budget sensitivity of BadWAM.** Increasing the query budget B raises the optimization cost, but also gives the optimizer more opportunity to find stronger perturbations. The perturbation-budget study further shows that larger ϵ consistently improves attack effectiveness.

tion reduces predicted-future distance from 14.01 to 13.04. Thus, a WAM can be driven toward failure while retaining a comparatively stable imagined future.

These dynamics clarify the trade-off: large action and future shifts indicate low stealth, whereas low future drift with weak action shifts indicates over-regularization. The default configuration balances both, producing damaging action changes with comparatively stable future predictions.

5.2 Transferability Across WAM Variants

To answer **RQ4**, Table 2 compares attacks transferred from a source WAM with the same objective optimized directly on the target WAM. Across three action-only transfer directions, transferred success is 64.1%–70.8%, only 2.6–4.7 percentage points (pp) above direct attacks, while still reducing target clean success by 27.6–34.0 pp. Imagination-preserving attacks also transfer between Joint and IDM WAMs, with transferred/direct success of 71.3%/67.0% and 65.0%/63.0% in the two directions. The corresponding future distances remain comparable (13.85/14.18 and 14.16/14.24). Across all five directions, transferred attacks are only 2.0–4.7 pp weaker, indicating shared observation-space vulnerabilities rather than action-head overfitting.

5.3 Ablations: Future Preservation, Perturbation Budget, and Runtime

To answer **RQ5**, we ablate the future-preserving weight, perturbation budget, and query budget. On LIBERO, Figure 7 shows that increasing λ reduces predicted-future drift but eventually weakens the attack. The preferred trade-offs are $\lambda = 0.03$ for the Joint WAM and $\lambda = 0.015$ for the IDM WAM, balancing induced failures and D_{img}. Stars denote representative nondominated points. Larger weights further preserve imagination but reduce action disruption, confirming that λ controls the stealth–strength trade-off. RoboTwin results are reported in Appendix B.6.5.

Figure 8 evaluates budget sensitivity on full LIBERO. Increasing ϵ has the largest effect, reducing Joint and IDM WAM success from about 97% to 63.0% and 67.0% at $\epsilon = 0.06$, and to 0% at $\epsilon = 0.20$. Larger query budgets provide smaller attack-strength gains with near-linear runtime cost.

The query-budget ablation reveals a runtime–effectiveness trade-off. Increasing B from 1 to 32 reduces success from 67.8% to 60.3% for the Joint WAM and from 72.1% to 65.9% for the IDM WAM, but increases per-replan runtime from about 2.6–2.8 s to 28.7–30.9 s. Most gains are achieved by $B = 16$; doubling the budget provides limited improvement at nearly twice the cost. We therefore use $B = 16$ by default.

6 Conclusion

World-action models couple action generation with imagined futures. In this work, we show that this coupling creates a new attack surface. We introduce BadWAM, a black-box framework that induces task failures in WAMs through small, bounded observation perturbations. Across multiple WAM variants and benchmarks, both BadWAM objectives, the action-only and imagination-preserving attacks, substantially reduce task success, while the latter provides a stealthier yet still effective attack. These findings show that reliable WAM safety requires verifying the consistency between imagined futures and executed actions, rather than trusting either interface in isolation.

AI USE STATEMENT

LLM tools were used to assist with manuscript drafting, language editing, and minor revisions to supporting visualization. All generated text and code changes were reviewed and verified by the authors, who take full responsibility for the final content and implementation.

ETHICS STATEMENT

This work studies BadWAM, a black-box observation-space attack against world-action models, with the goal of identifying safety vulnerabilities and informing more robust WAM design. All experiments are conducted in simulation; no perturbations are deployed on physical robots, and the study involves no human subjects or personal data. The attack should be used only for authorized security evaluation and safety research.

REPRODUCIBILITY STATEMENT

We provide detailed descriptions of the threat model, attack objectives, perturbation and query budgets, closed-loop evaluation protocol, model configurations, training procedures, datasets, metrics, and ablations in the paper and appendices. Experiments use the publicly available LIBERO and RoboTwin benchmarks and officially released action-only checkpoints; the other WAM variants are trained with the stated default settings. We will release the attack implementation and configurations to facilitate reproduction of the reported results.

REFERENCES

Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, et al. pi_0: A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 2024.

Tom B. Brown, Dandelion Mané, Aurko Roy, Martín Abadi, and Justin Gilmer. Adversarial patch. *arXiv preprint arXiv:1712.09665*, 2017.

Linghan Chen, Kaiyan Ji, and Minyu Guo. Attacking the trusted imagination: Oracle-level integrity attacks on imagine-then-act world models. *arXiv preprint arXiv:2606.22966*, 2026a.

Pin-Yu Chen, Huan Zhang, Yash Sharma, Jinfeng Yi, and Cho-Jui Hsieh. Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models. In *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*, pp. 15–26, 2017. doi: 10.1145/3128572.3140448.

Ronghan Chen, Yandan Yang, Zuojin Tang, Dongjie Huo, Tong Lin, Haoning Wu, Haoyun Liu, Yuzhi Chen, Lulu Zheng, Botai Yuan, Tianlun Li, Dekang Qi, et al. Abot-m0.5: Unified mobility-and-manipulation world action model. *arXiv preprint arXiv:2607.00678*, 2026b.

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

Kevin Eykholt, Ivan Evtimov, Earlence Fernandes, Bo Li, Amir Rahmati, Chaowei Xiao, Atul Prakash, Tadayoshi Kohno, and Dawn Song. Robust physical-world attacks on deep learning visual classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1625–1634, 2018. doi: 10.1109/CVPR.2018.00175.

Adam Gleave, Michael Dennis, Cody Wild, Neel Kant, Sergey Levine, and Stuart Russell. Adversarial policies: Attacking deep reinforcement learning. In *International Conference on Learning Representations*, 2020.

Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations*, 2015.

Zhixiang Guo, Siyuan Liang, András Balogh, Noah Lunberry, Rong-Cheng Tu, Márk Jelasity, and Dacheng Tao. When world models dream wrong: Physical-conditioned adversarial attacks against world models. *arXiv preprint arXiv:2602.18739*, 2026.

Andrew Ilyas, Logan Engstrom, Anish Athalye, and Jessy Lin. Black-box adversarial attacks with limited queries and information. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 2137–2146. PMLR, 2018.

Physical Intelligence, Bo Ai, Ali Amin, Raichelle Aniceto, Ashwin Balakrishna, Greg Balke, Kevin Black, George Bokinsky, Shihao Cao, Thomas Charbonnier, et al. $\pi_{0.7}$: A steerable generalist robotic foundation model with emergent capabilities. *arXiv preprint arXiv:2604.15483*, 2026.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan P Foster, Pannag R Sanketi, Quan Vuong, et al. Openvla: An open-source vision-language-action model. In *Conference on Robot Learning*, pp. 2679–2713. PMLR, 2025.

Qi Li, Bo Yin, Weiqi Huang, Ruhao Liu, Bojun Zou, Runpeng Yu, Jingwen Ye, Weihao Yu, and Xinchao Wang. Vision-language-action safety: Threats, challenges, evaluations, and mechanisms. *arXiv preprint arXiv:2604.23775*, 2026.

Yen-Chen Lin, Zhang-Wei Hong, Yuan-Hong Liao, Meng-Li Shih, Ming-Yu Liu, and Min Sun. Tactics of adversarial attack on deep reinforcement learning agents. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pp. 3756–3762, 2017.

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, 2023.

Hanqing Liu, Songping Wang, Jiahuan Long, Jiacheng Hou, Jialiang Sun, Chao Li, Yang Yang, Wei Peng, Xu Liu, Tingsong Jiang, et al. Jailwam: Jailbreaking world action models in robot control. *arXiv preprint arXiv:2604.05498*, 2026a.

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*, 2026b.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, 2018.

Utkarsh A. Mishra, Yongxin Chen, Danfei Xu, Yang Liu, Xi Chen, and Jiayuan Mao. Understanding and mitigating the video-action generalization gap via temporal ratio. *arXiv preprint arXiv:2607.08127*, 2026.

Linghui Shen, Mingyue Cui, and Xingyi Yang. Badworld: Adversarial attacks on world models. *arXiv preprint arXiv:2606.16519*, 2026a.

Qiuhong Shen, Shihua Zhang, Yue Liao, Qi Li, Zhenxiong Tan, Shizun Wang, Shuicheng Yan, and Xinchao Wang. World action models: A survey. *arXiv preprint arXiv:2606.20781*, 2026b.

James C. Spall. Multivariate stochastic approximation using a simultaneous perturbation gradient approximation. *IEEE Transactions on Automatic Control*, 37(3):332–341, 1992. doi: 10.1109/9.119632.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In *International Conference on Learning Representations*, 2014.

Shuai Tian, Yupeng Zheng, Yuhang Zheng, Songen Gu, Yujie Zang, Yuxing Qin, Weize Li, Haoran Li, Wenchao Ding, and Dongbin Zhao. Vt-wam: Visual-tactile world action model for contact-rich manipulation. *arXiv preprint arXiv:2607.02503*, 2026.

Jonathan Uesato, Brendan O'Donoghue, Aaron van den Oord, and Pushmeet Kohli. Adversarial risk and the dangers of evaluating against weak attacks. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 5025–5034. PMLR, 2018.

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.

Wentao Zhao, Jiaming Chen, Ziyu Meng, DonghuiMao, Ran Song, and Wei Zhang. VLMPC: Vision-Language Model Predictive Control for Robotic Manipulation. In *Proceedings of Robotics: Science and Systems*, Delft, Netherlands, July 2024. doi: 10.15607/RSS.2024.XX.106.

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

A Appendix

B Details of Experiment Setup

B.1 Training Details

For the action-only WAM, we use the official FastWAM checkpoints (Yuan et al., 2026). The joint WAM and IDM WAM variants used in our LIBERO and RoboTwin experiments are trained from the FastWAM release using the default training configuration. All the models are trained on 8 H100 GPUs. Figures 9 and 10 summarize the optimization trajectories of the RoboTwin Joint and IDM WAMs, respectively.

The training setup is given in Table 3. For LIBERO, the model uses two cameras, namely the third-person view and the wrist view. Each camera is resized to 224×224, and the two camera streams are concatenated horizontally. For RoboTwin, following the official FastWAM preprocessing, we compose the three camera views into a single 384×320 frame. The high-camera view is resized to 256×320 and placed above the two wrist views, each resized to 128×160 and concatenated horizontally.

The model backbone and tokenizer follow FastWAM. Specifically, the configuration uses `Wan-AI/Wan2.2-TI2V-5B` as the video backbone and `Wan-AI/Wan2.1-T2V-1.3B` as the tokenizer model. We keep the default FastWAM action and video schedulers. We do not modify the FastWAM architecture when training the joint and IDM variants.

B.2 Attack Hyperparameters and Query Accounting

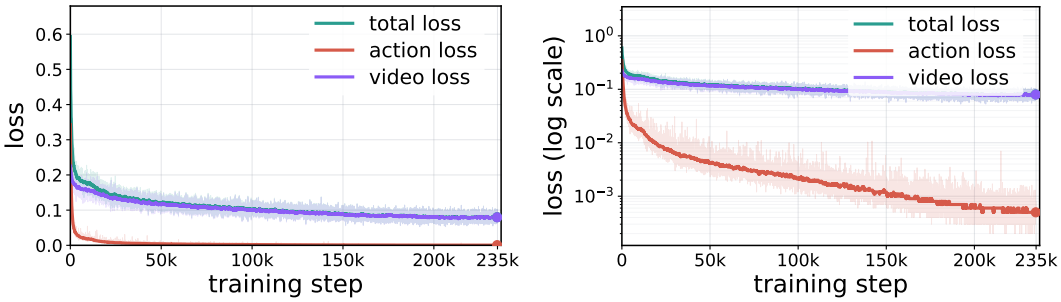
Unless otherwise stated, BadWAM perturbs the full preprocessed visual input under an ℓ_∞ bound of $\epsilon = 0.06$. The perturbation is applied online at each replanning step during closed-loop execution. The default zeroth-order finite-difference search uses 16 perturbation queries plus one clean reference query per attacked replan, for a total of 17 WAM forward queries per attacked replan. Internally, the 16 perturbation queries are organized as 8 paired finite-difference updates. Each update samples one random Rademacher direction, evaluates positively and negatively perturbed inputs, and updates the perturbation by a projected signed step. We use a finite-difference radius of 0.02 and a search step size of 0.02.

For the action-only adversarial attack, the future-preserving coefficient is set to zero and the objective only maximizes clean0attacked action deviation. For the imagination-preserving adversarial attack, the objective additionally penalizes predicted-future drift. The default full-sweep setting uses $\lambda = 0.015$ for the future-preserving coefficient. The matched-strength analysis in Figure 6 uses a larger future-preserving coefficient when comparing the two objectives under similar attack strength, so that the effect of the preservation term is easier to isolate.

Table 3: **Training configurations of the four WAMs.** "Video input" denotes the spatial resolution after composing all camera views, and d_a/d_p denotes the action/proprioception dimensions. "Seq./horizon" reports the number of frames and the corresponding action horizon. The effective batch size is $B_{\text{eff}} = B_{\text{GPU}} \times 8 \times \text{Acc}$.

Dataset	Model	Views	Video input	d_a/d_p	Seq./horizon	Epochs	Opt. steps	B_{GPU}	Acc.	B_{eff}	Normalization	Grad. ckpt.
LIBERO	Joint WAM	2	224×448	7/8	33/32	10	21,700	16	1	128	Min0max	No
LIBERO	IDM WAM	2	224×448	7/8	33/32	10	21,700	8	2	128	Min0max	Yes
RoboTwin	Joint WAM	3	384×320	14/14	33/32	5	234,830	16	1	128	Z-score	No
RoboTwin	IDM WAM	3	384×320	14/14	33/32	5	234,830	16	1	128	Z-score	Yes

All models are initialized from the same Wan2.2-TI2V-5B and ActionDiT backbones and trained on eight NVIDIA H100 80GB GPUs using DeepSpeed ZeRO-1 and BF16 precision. We use AdamW with $\beta_1 = 0.9$, $\beta_2 = 0.95$, learning rate 10^{-4}, weight decay 10^{-2}, and gradient clipping at 1.0. The learning rate follows a cosine schedule after a 5% linear warm-up and decays to 10^{-6}. We set the action-loss weight to 1.0, the action-to-video frequency ratio to 4, and the random seed to 42.



(a) Linear scale. (b) Log scale.

Figure 9: **RoboTwin Joint WAM training loss.** The linear-scale view shows the overall convergence trend, while the log-scale view makes the continued decay of the action loss visible. Light curves denote raw logged losses and bold curves denote smoothed losses.

B.3 White-Box Gradient-Access Reference

Figure 3 includes a white-box gradient-access reference to contextualize the strength of our query-based attacks. This reference is not part of the main BadWAM threat model. BadWAM assumes query access and does not require gradients or model parameters. The white-box reference instead asks how much stronger the attack can become if the adversary can backpropagate through the WAM's action output.

At the beginning of an episode, the attacker optimizes one bounded input perturbation using projected gradient ascent. The objective maximizes the deviation between the clean action chunk and the perturbed action chunk. The perturbation is initialized randomly inside the ℓ_∞ ball, updated for 16 gradient steps with step size 0.01, and projected back to $\|\delta\|_\infty \leq 0.06$ after each step. After the perturbation is found, the policy is evaluated with the standard closed-loop inference procedure. No model weights are modified, and the environment dynamics are not differentiated through.

This reference serves two purposes. First, it separates generic visual fragility from targeted action manipulation: random perturbations under the same budget are substantially weaker than targeted attacks. Second, it shows that gradient access can further amplify the vulnerability, which suggests that the query-based BadWAM attacks are not saturating the worst-case attack strength. Since this reference uses

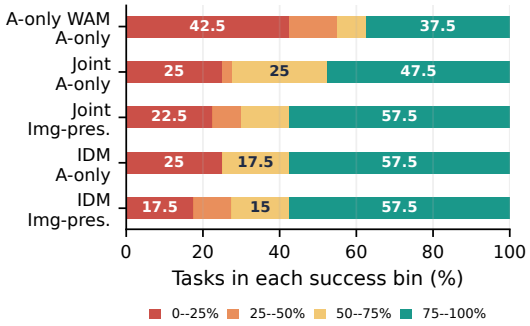
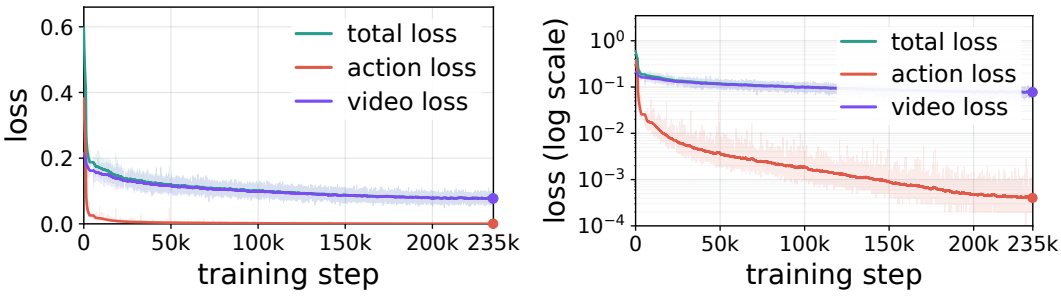


Figure 11: **Task-level failure profile under attack on LIBERO.** This figure groups the 40 LIBERO tasks according to their attacked closed-loop success rates, using 20 trials per task. Lower bins indicate tasks that are consistently broken rather than only mildly degraded.



(a) Linear scale.

(b) Log scale.

Figure 10: **RoboTwin IDM WAM training loss.** The linear-scale view shows the overall convergence trend, while the log-scale view makes the continued decay of the action loss visible. Light curves denote raw logged losses and bold curves denote smoothed losses.

stronger access than BadWAM, we treat it as a diagnostic upper-bound reference rather than as the primary threat model.

B.4 Random Perturbation Baseline

The random perturbation baseline uses the same ℓ_∞ budget as BadWAM but removes the optimization objective. For each attacked observation, it samples an independent uniform perturbation from $[-\epsilon, \epsilon]$ and clips the perturbed input to the valid model input range. This baseline is model-agnostic and does not use action outputs, future predictions, gradients, or task feedback. Its role is to test whether task failures are caused merely by arbitrary visual corruption. The results in Figure 3 show that optimized world-action drift attacks are consistently stronger than random perturbations under the same visual budget.

B.5 Evaluation Metrics

Our primary metric is closed-loop task success. Lower success under attack indicates a stronger attack. In addition to success rate, we report action distance, predicted-future distance, and input perturbation statistics. The action distance is computed as the mean ℓ_2 deviation between clean and attacked action chunks. The predicted-future distance is computed analogously on the future representation or decoded future-video tensor available through the WAM interface. The decoupling score is the ratio between action distance and predicted-future distance, with a small numerical constant in the denominator for stability. A high decoupling score indicates that the attack obtains a larger action shift per unit of predicted-future drift.

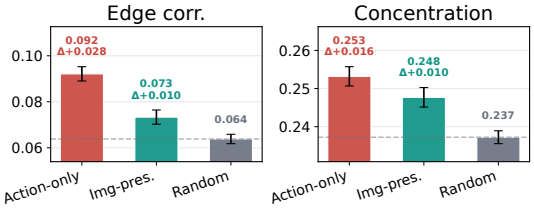


Figure 12: **Perturbation fingerprints on RoboTwin Joint WAM.** The ℓ_∞ perturbations is $\epsilon = 0.06$ and query budget $B = 16$ per replan. Compared with random noise, both attacks are more edge-aligned and spatially concentrated. Error bars denote 95% confidence intervals.

For repeated-trial evaluation, we use pass@k. For a fixed task, pass@k is the fraction of successful executions among the first k trials. We then average this quantity across tasks. This metric shows whether an attack only causes isolated unlucky failures or consistently lowers reliability as more trials are considered.

B.6 More Results

B.6.1 Systematic Patterns of Failures.

Here, we test whether the success drop survives repeated trials. Figure 15 plots pass@k as k grows from 1 to 20. If BadWAM only exploited a few unlucky seeds, the attacked curves would move back toward the clean curves as more trials are averaged. Instead, the separation persists across the full

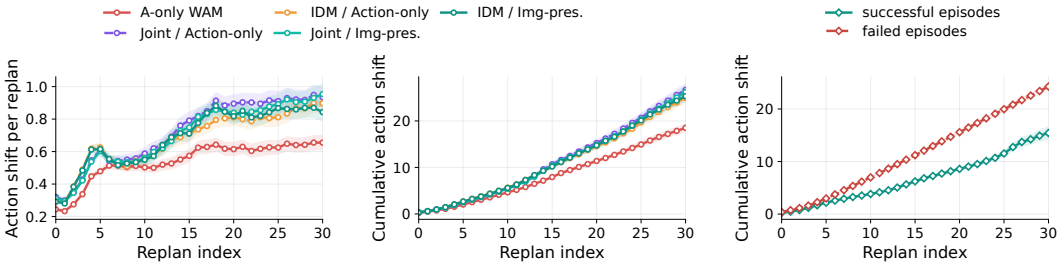


Figure 13: **Per-replan action shifts persist across execution and accumulate over time.** Failed episodes showing substantially larger cumulative shifts than successful ones. Action shift is measured between clean and attacked action chunks at each replan.

trial. The clean action-only WAM stays close to 1.0, while the attacked action-only WAM remains around 0.40-0.43. The joint and IDM WAMs also remain consistently below their clean references under both attack objectives.

B.6.2 Task-Level Failure Distribution.

Figure 11 complements the aggregate table with a task-level failure profile. For the action-only WAM, 42.5% of tasks fall into the 0-25% success bin under attack. For the joint and IDM WAMs, 17.5-25.0% of tasks fall into this lowest-success bin, depending on the objective. This distributional view is important for safety: a moderate average can hide tasks that become almost unusable.

B.6.3 Progressive Closed-Loop Drift.

This example highlights why closed-loop evaluation is necessary. The perturbation need not make the observation obviously nonsensical or cause an obviously catastrophic first action. Instead, it can introduce structured deviations that remain locally plausible but become physically consequential over the horizon. Figure 13 confirms this behavior at the dataset level: per-replan action shifts persist across execution and accumulate over time. Failed episodes accumulate substantially larger action shifts than successful ones, suggesting that BadWAM failures often emerge from progressive closed-loop drift rather than a single isolated bad action.

B.6.4 Spatial Structure of the Perturbations.

We further examine whether BadWAM produces structured input perturbations rather than merely exploiting generic image noise. For each replan, *edge correlation* is the Pearson correlation between per-pixel mean absolute perturbation magnitude and the clean image's gradient magnitude; *concentration* is the Gini coefficient of the spatial perturbation-magnitude map. As shown in Figure 12, both objectives are more aligned with image edges and more spatially concentrated than random perturbations.

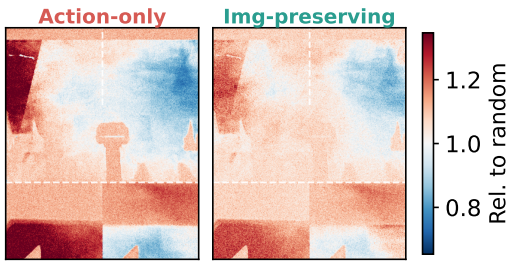


Figure 14: **Relative perturbation fingerprints on RoboTwin Joint WAM.** Both BadWAM attacks use full-image ℓ_∞ perturbations with $\epsilon = 0.06$ and query budget $B = 16$ per replan. Colors show perturbation strength relative to random noise: red indicates stronger-than-random perturbation energy and blue indicates weaker-than-random energy. Dashed lines separate the composed RoboTwin camera views.

Figure 14 further localizes this structure relative to the random baseline. Both objectives redistribute perturbation energy non-uniformly across the composed camera views, amplifying selected regions while suppressing others. The action-only objective produces stronger spatial contrast, whereas the imagination-preserving objective exhibits a milder but related pattern. Since all methods use the same full-image ℓ_∞ constraint, these patterns arise from the attack optimization rather than from a hand-designed spatial mask.

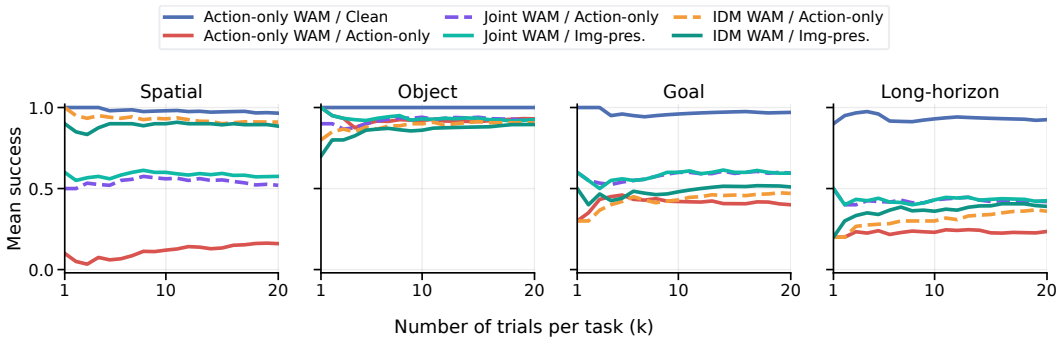


Figure 15: **Mean pass@k across trials on different LIBERO suites.** BadWAM lowers success throughout the trial budget instead of only causing isolated unlucky failures.

B.6.5 RoboTwin Action-Imagination Trade-off.

We repeat the future-weight ablation on the RoboTwin Joint WAM to test whether the trade-off also appears beyond LIBERO. As shown in Figure 16, evaluated on a 12-task RoboTwin subset with 10 episodes per task, the default $\lambda = 0.015$ achieves 85.8% task success under attack, compared with 94.2% for clean execution and 84.2% for the action-only objective. Relative to $\lambda = 0$, its action and future distances decrease to 0.86 and 0.96, respectively. As λ increases further, the action shift decreases steadily, reaching 0.71 at $\lambda = 0.3$, while future drift remains near 0.96 and closed-loop success varies between 84.2% and 88.3%. Thus, the action-imagination trade-off also appears on RoboTwin but with different patterns.

B.7 What Do These Results Imply for WAM Safety?

Across **RQ1-RQ6**, the results point to a subtle failure mode for imagination-based safety checks. The relevant safety property is not the predicted-future output in isolation, but synchronization between the imagined future and the action that will actually be executed.

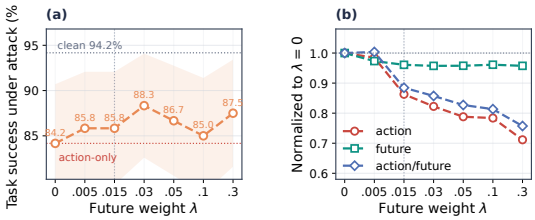


Figure 16: **Effect of the future-preserving weight on the RoboTwin Joint WAM. (Left)** Task success under attack; lower is stronger. **(Right)** Action & future drift, with ratio normalized to $\lambda = 0$. Larger λ reduces action shift while preserving future drift, revealing the strength–stealth trade-off.

This also explains why the attack is WAM-specific. For a conventional action-only policy, an adversarial perturbation can only be assessed through its effect on the action or final task outcome. For a WAM, the attacker can exploit a gap between two interfaces: the future-imagination interface and the action interface. The action-only BadWAM objective targets the high-strength endpoint of this problem; the imagination-preserving objective targets a stealthier endpoint. The latter is especially relevant for systems that expose predicted futures to planners, runtime monitors, or human supervisors. Even when the predicted future remains visually close to the clean prediction, the executed action chunk may no longer be the action that would realize that future.

Finally, the transfer results suggest that the vulnerability is not merely an implementation artifact. Perturbations optimized on one WAM variant remain damaging on another, indicating shared observation-space directions that affect action generation.