AWARE: ACTION WEAK-SPOT ASSESSMENT BEFORE ROBOTIC-WORLD-MODEL EXECUTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Action-conditioned world models serve as environment surrogates for robotic planning, policy evaluation, and control, yet visually plausible predictions do not guarantee accurate modeling of action effects. Small action perturbations can induce response errors that propagate across successive prediction windows and ultimately produce substantial prediction drift. Existing visual and textual attacks conflate perceptual, semantic, and dynamics-prediction errors, making action-response errors difficult to isolate. We study the action-conditioning input as a separate attack surface and formalize pre-deployment action weak-spot discovery: identifying directions likely to induce large action-response mismatch between a world model and its environment without access to future environment responses. We propose AWARE (Action Weak-spot Assessment before Robotic-world-model Execution). Holding observation history and task semantics fixed, AWARE constructs a candidate pool under a shared budget in normalized action space and selects the candidate with the largest frozen-model response energy relative to the nominal prediction. Local response-geometry analysis characterizes the directional-alignment conditions and information limits under over-response and under-response. Compared with same-pool random selection, AWARE achieves higher mean true action-response mismatch at every tested budget: relative increases of 46.8–68.9% on CALVIN/Ctrl-World, 67.6–99.9% on RoboCasa/Ctrl-World, and 118.4–1473.1% on CALVIN/IRA-SIM. AWARE provides a diagnostic of action–outcome consistency before deployment. Exploratory repair improves selected response measures but does not consistently reduce mismatch on a fixed episode-disjoint development set.

1 INTRODUCTION

Action-conditioned world models simulate the consequences of candidate actions for robotic planning, policy evaluation, and model-based control (Zhu et al., 2025; Guo et al., 2025). As downstream systems increasingly rely on these predictions to select actions, reliable operation depends on how faithfully the model captures action effects. We investigate whether the action-conditioning input itself constitutes an independent attack surface for world models.

Existing attacks typically perturb visual observations or language instructions, altering the model's understanding of the scene or task (Shen et al., 2026; Li et al., 2026; Liu et al., 2026). Visual attacks exploit latent representations or generative dynamics to amplify bounded contextual perturbations; textual attacks use adversarial instructions or jailbreak prompts to redirect predicted trajectories or actions. These attacks can induce prediction collapse, trajectory deviation, or dangerous behavior. However, their effects conflate perceptual, semantic, and dynamics-prediction errors, making it difficult to isolate an incorrect correspondence between actions and physical outcomes.

Visual and textual attacks probe how predictions change when state representations or task semantics are altered. Action reliability also requires the model and environment to respond consistently to different actions. Small perturbations to action-conditioning inputs initially produce only subtle differences between predicted and actual responses. Across successive prediction windows, however, initial response errors propagate and are amplified, leading to substantial prediction drift. Because visual and textual perturbations also alter the state or semantic reference, they cannot readily isolate these action-response errors from perceptual and semantic errors.

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 summarizes the action-response failure mode, the AWARE selection procedure, and the subsequent repair experiment.

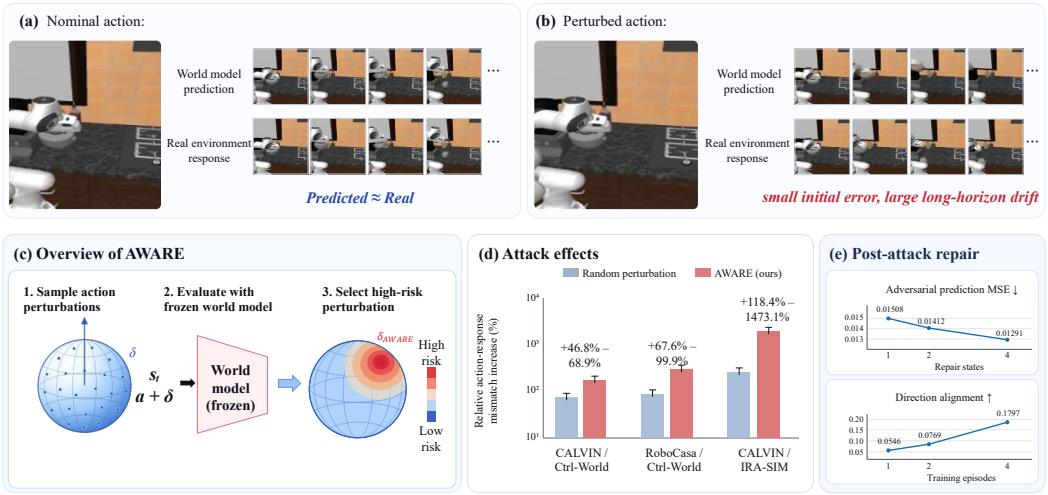


Figure 1: Overview of action weak-spot discovery and post-attack repair. (a,b) Nominal and perturbed action sequences illustrate possible prediction drift. (c) AWARE selects the highest-scoring action perturbation using a frozen world model. (d) Attack effects across model–environment pairs. (e) An illustrative repair result.

We therefore perturb only the action and compare local action-response geometries. Within a state–action neighborhood, world-model response energy and true action-response mismatch admit quadratic approximations; alignment between their directional geometries determines whether they rank perturbations consistently. Figure 2 illustrates two typical modes of action-response mismatch between an action-conditioned world model and its environment: *over-response*, where the model overestimates action effects, and *under-response*, where it underestimates them. Our analysis shows when model responses can reveal weak spots under over-response and when they can remain informative under under-response if the model preserves the direction of the environment's action response.

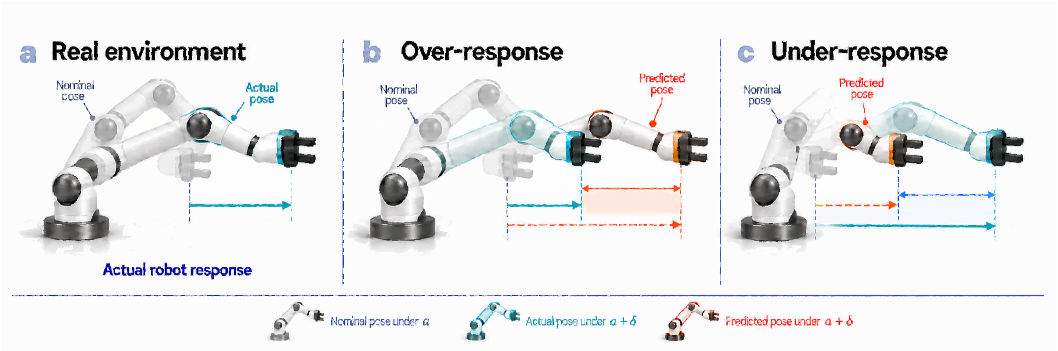


Figure 2: Two typical mismatch regimes under the same action perturbation. (a) Action response in the environment. (b) The predicted action response exceeds the real response (over-response). (c) The predicted action response is smaller than the real response (under-response).

Based on these principles, we propose **A**ction **W**eak-spot **A**ssessment before **R**obotic-world-model **E**xecution (AWARE), a framework for discovering action weak spots in robotic world models before deployment. AWARE samples bounded perturbations in normalized action space, maps perturbed actions into the admissible range, and selects the candidate with the largest frozen-model response relative to the nominal action. The experiments use variable-radius pools; the fixed-radius

construction is used for the local analysis. This converts continuous optimization into finite candidate ranking under a shared budget. Environment responses are used only for offline mismatch evaluation, never for candidate generation, scoring, or selection.

We compare AWARE with a same-pool random control using identical states, action budgets, and candidate pools; only the selection rule differs. Across four budgets, AWARE increases true mismatch by 46.8%, 54.1%, 68.9%, and 52.9% on CALVIN/Ctrl-World; by 99.9%, 98.7%, 79.9%, and 67.6% on RoboCasa/Ctrl-World, which changes the tasks and action space; and by 487.1%, 1473.1%, 389.4%, and 118.4% on CALVIN/IRA-SIM, which changes the model architecture and action-injection mechanism. These simulation results show positive mean gains in the three tested environment–model settings. They do not establish reliability on physical robots or across world-model architectures beyond Ctrl-World and IRA-SIM.

Our main contributions are as follows:

1. We isolate the action-conditioning input as an attack surface and formalize pre-deployment action weak-spot discovery.
2. We introduce AWARE, which builds a randomized candidate pool, ranks actions by the frozen world model's relative response, and selects the highest-scoring perturbation.
3. Experiments across distinct world-model/environment combinations show that AWARE can identify action weak spots. Exploratory post-attack repair shows that the selected adversarial examples can improve aspects of world-model response magnitude and direction.

2 Related Work

Physics-based simulators. Physics-based simulators provide essential infrastructure for robot learning and embodied AI. MuJoCo, Isaac Gym, RLBench, Meta-World, SAPIEN, ManiSkill2, and related platforms provide controllable environments for policy learning, planning, and safety evaluation by explicitly modeling geometry, dynamics, contacts, sensors, and tasks (Todorov et al., 2012; Makoviychuk et al., 2021; James et al., 2020; Yu et al., 2020; Xiang et al., 2020; Gu et al., 2023). They support reproducible experiments and controlled interventions but require manually specified assets, contact models, physical parameters, and task distributions. We use executable simulation not as a replacement for learned models, but as a counterfactual reference for testing whether a learned simulator follows the commanded action.

World models as learned simulators. World models learn environment transitions from data and generate imagined rollouts for planning, policy evaluation, and counterfactual reasoning. This idea appears in Dyna and later pixel-based latent dynamics models such as World Models, PlaNet, and Dreamer (Sutton, 1991; Ha & Schmidhuber, 2018; Hafner et al., 2019; 2023). Recent generative simulators, including UniSim, GAIA-1, Genie, RoboDreamer, IRASim, and Ctrl-World, extend this approach to visually rich, action-conditioned prediction (Yang et al., 2023; Hu et al., 2023; Bruce et al., 2024; Zhou et al., 2024; Zhu et al., 2025; Guo et al., 2025). For such models, realism is not enough: a rollout must also represent how an action changes the scene. Recent evaluations show that robotic video world models still fail on spatial reasoning, contact prediction, physical deformation, hallucination, and action controllability (Jiang et al., 2026; Hansen & Wang, 2026). These failures motivate evaluating action responses in addition to visual fidelity.

Attacks and robustness of world models. Adversarial learning has shown that small or structured perturbations can change model behavior, motivating robust optimization, black-box attacks, and physical-world attacks (Szegedy et al., 2014; Goodfellow et al., 2015; Carlini & Wagner, 2017; Madry et al., 2018; Ilyas et al., 2018; Athalye et al., 2018). In sequential decision-making and embodied settings, attacks may target observations, dynamics, adversarial policies, or interaction processes (Huang et al., 2017; Pinto et al., 2017; Gleave et al., 2020). Recent work affects world models and robot control through perturbations to physical conditions, visual context, attack configurations, or high-level instructions (Guo et al., 2026a; Shen et al., 2026; Guo et al., 2026b; Li et al., 2026; Liu et al., 2026). BadWAM and our work both examine desynchronization between imagined outcomes and action effects, but BadWAM perturbs the visual input of a world–action model and evaluates closed-loop task failure; its "action-only" variant specifies an attack target rather than a perturbation

to the commanded-action input (Li et al., 2026). We hold visual context and task semantics fixed and perturb only the current commanded action fed to a predictive world model. Selection uses no future environment outcomes; offline evaluation measures the difference between model and environment action responses. Table 1 summarizes these interface differences. A common benchmark would require the same model interface, current-action intervention, candidate budget, and paired environment-response metric; published attack success rates cannot be compared directly with our mismatch MSE.

Table 1: Differences between AWARE and prior work.

Study	Changed input or search object	Evaluation target
BadWorld (Shen et al., 2026)	Context image	Future-rollout degradation under controls
WMAttack (Guo et al., 2026b)	Attack configuration	Reward degradation of world-model agents
BadWAM (Li et al., 2026)	Visual input to world-action model	Closed-loop action failure and imagination drift
JailWAM (Liu et al., 2026)	High-level instruction	Unsafe behavior in robot control
AWARE	Current commanded action	Model–environment action-response mismatch

3 METHOD

Given a state or observation s, nominal action a, and budget b, AWARE uses only predictions from a frozen world model to select a perturbation likely to expose action-response mismatch with the environment. Figure 3 summarizes the pre-deployment selection procedure.

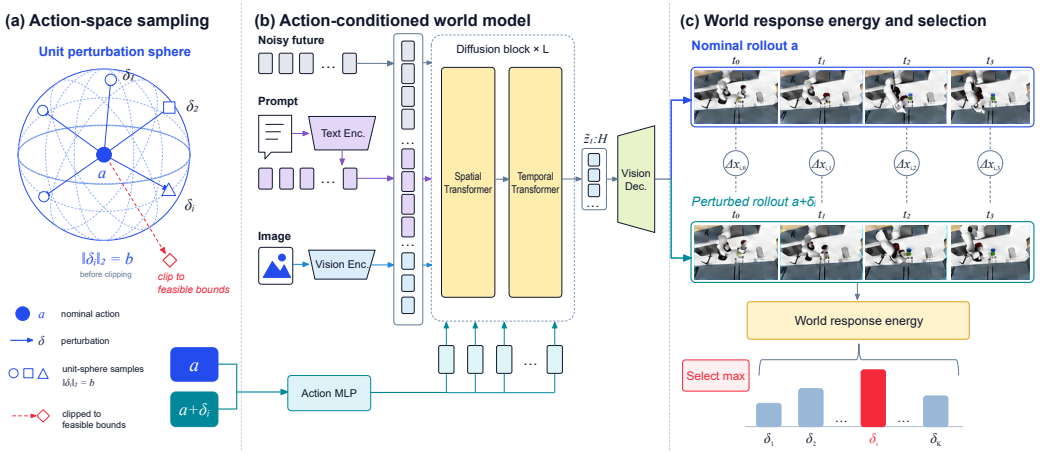


Figure 3: Overview of AWARE. (a) Generate perturbation candidates in normalized action space and map them into the admissible action range. (b) Hold the context and world model fixed and query rollouts for nominal and candidate actions. (c) Compute each candidate's world-response energy relative to the nominal prediction and select the highest-scoring candidate.

3.1 PROBLEM DEFINITION

We study an independent attack surface in action-conditioned robot world models: the action-conditioning input. Given the current state or observation s and a nominal action a, the attacker perturbs the current one-step action as $a^{\text{adv}} = a + \delta$.

To evaluate whether an action perturbation exposes a world-model weakness, we use action-response mismatch as the offline objective:

$$M(s, a, \delta) = \frac{1}{D} \left\| \Delta x_W(s, a, \delta) - \Delta x_E(s, a, \delta) \right\|_2^2, \tag{1}$$

where $\Delta x_Y(s, a, \delta) = x_Y(s, a+\delta) - x_Y(s, a)$ for $Y \in \{W, E\}$. Here, $x_E(s, a)$ denotes the response produced by the real environment under action a, $x_W(s, a)$ denotes the world-model prediction of that response, and D is the number of scalar elements in the response tensor used by the metric.

3.2 Constructing the Attack Candidate Pool

Gradient-based optimization of the world-model response may follow directions that do not increase true mismatch (Section 4.3; Appendix A.1). We instead construct a finite candidate pool $\Delta_s(b) = \{\delta_i\}_{i=1}^K$ for state s, nominal action a, and budget b. For the fixed-radius construction, sample $u_i \sim \mathrm{Unif}(\mathbb{S}^{d-1})$ in normalized action space and set $\delta_i = bu_i$, where $\mathbb{S}^{d-1}$ is the unit sphere. The variable-radius distributions used in the experiments are specified in Appendix B.2. Normalization removes scale differences across action dimensions so that larger numerical ranges do not dominate perturbation directions.

The perturbed action $a + \delta_i$ may violate the lower or upper bounds of the environment action space. We therefore map it into the admissible space $\mathcal{A}$ and define the final candidate perturbation as the difference between the resulting executable action and the nominal action. This construction converts continuous perturbation search into finite candidate selection under the same action budget.

3.3 Selecting the Best Perturbation from the Candidate Pool

Future environment responses are usually unavailable before execution, so $M(s, a, \delta)$ cannot be computed directly. We instead seek high-mismatch perturbations indirectly through world-model predictions. For $\delta \in \Delta_s(b)$, define the world-model response energy as

$$Q_W(s, a, \delta) = \frac{1}{D} \|\Delta x_W(s, a, \delta)\|_2^2 . \tag{2}$$

This score measures the model's local sensitivity to the action perturbation: a large response to a small perturbation indicates a strong predicted action effect. AWARE uses it as a pre-deployment proxy for true action-response mismatch.

Finally, AWARE selects the perturbation with the largest world-model response energy from the candidate pool:

$$\delta_{\mathrm{AWARE}}(s, b) = \arg\max_{\delta \in \Delta_s(b)} Q_W(s, a, \delta). \tag{3}$$

3.4 Theoretical Analysis

We use local linearization to characterize when world-model response energy can serve as a proxy for true mismatch. Around a fixed state–action pair, $\Delta x_W \approx J_W \delta$ and $\Delta x_E \approx J_E \delta$, where J_W and J_E are the corresponding action Jacobians. Omitting the common factor D^{-1}, which does not affect rankings, define the local quadratic scores

$$\begin{aligned} q(\delta) &= \delta^\top A \delta, \quad A = J_W^\top J_W, \\ m(\delta) &= \delta^\top B \delta, \quad B = (J_W - J_E)^\top (J_W - J_E). \end{aligned} \tag{4}$$

Thus, $q(\delta) \approx DQ_W(s, a, \delta)$ and $m(\delta) \approx DM(s, a, \delta)$; the covariance below concerns these quadratic approximations.

For $\delta = bu$ with $u \sim \mathrm{Unif}(\mathbb{S}^{d-1})$, define $A_0 = A - \mathrm{tr}(A)I/d$ and $B_0 = B - \mathrm{tr}(B)I/d$. The spherical fourth moment gives

$$\mathrm{Cov}\big(q(bu), m(bu)\big) = \frac{2b^4}{d(d+2)} \langle A_0, B_0 \rangle_F . \tag{5}$$

Positive alignment, $\langle A_0, B_0 \rangle_F > 0$, means that the local scores are positively correlated across directions. It does not by itself guarantee that their maximizers agree or that top-score selection improves mismatch in every pool. Appendix A.1 characterizes the expected gain from finite candidate selection, and Appendix A.3 derives Eq. (A.12) and bounds the local approximation error.

In the *over-response regime*, the world model overestimates action effects. When environment-response terms are small, $B \approx A$ and $\langle A_0, B_0 \rangle_F \approx \|A_0\|_F^2$. Alignment is strictly positive if $A_0 \neq 0$ and the cross terms are sufficiently small relative to this anisotropic component (Appendix A.3). Directions with strong model responses then tend to produce large true mismatch, providing a selection signal for AWARE.

In the *under-response regime*, write $J_W \approx J_E R$, where R is a direction-dependent action-space response operator, and let $S = J_E^\top J_E$. The two geometries become

$$A \approx R^\top S R, \qquad B \approx (R - I)^\top S (R - I). \tag{6}$$

For uniform attenuation $R = \alpha I$, $0 < \alpha < 1$, both geometries are positive multiples of S: the model underestimates response magnitudes but preserves sensitive directions, so their rankings agree. With direction-dependent attenuation, however, A and B need not align, and Q_W may misrank candidates. The representation $J_W = J_E R$ alone does not ensure attenuation in every direction; the precise condition is given in Appendix A.2.

Without environment feedback, no pre-deployment selector can guarantee a correct ranking under arbitrary direction-dependent under-response. Appendix A.2 establishes this information limit and gives illustrative examples.

4 Experiments

4.1 Experimental Setup

Datasets and Models. The main experiments use CALVIN and a subset of RoboCasa with their corresponding Ctrl-World checkpoints (Mees et al., 2022; Nasiriany et al., 2024; Guo et al., 2025). CALVIN provides manipulation trajectories, observations, and executed actions. We construct candidate perturbations at 90 fixed evaluation states from the test trajectories. Actions use the same normalization in the world model and simulator so that model inputs match executed actions.

The cross-model experiment uses IRA-SIM trained from scratch on a CALVIN-D subset (Zhu et al., 2025). We retain the CALVIN simulator, state set, action-perturbation settings, and evaluation protocol, replacing only the world model to distinguish model effects from environment effects.

Baselines and Ablations. We compare AWARE with a same-pool random control using identical states, perturbed dimensions, budgets, candidate pools, and rollout lengths. Only selection differs: AWARE maximizes world-model response, whereas the control selects randomly from the same pool.

We additionally consider the following ablations:

1. **Absolute World-Model Energy.** This ablation retains the candidate-generation and selection procedures of AWARE but scores each action by its absolute world-model response energy instead of the response change relative to the nominal prediction. It tests the role of relative response energy.
2. **Continuous AWARE.** This ablation replaces finite candidate ranking with continuous optimization of world-model response MSE relative to the nominal prediction, subject to the same action budget. It tests whether world-model response can directly guide continuous action optimization.

Metrics. All experiments use the action-response mismatch MSE defined in Section 3.1 as the primary evaluation metric. In addition to the mean mismatch of the selected actions, we report their relative improvement over the same-pool random control:

$$\text{Lift} = \frac{\text{MSE}_{\text{selected}} - \text{MSE}_{\text{random}}}{\text{MSE}_{\text{random}}}. \tag{7}$$

Each setting and budget uses the same 90 states. The archived main experiments average eight distinct same-pool random actions per state in all three settings. Figure 4 summarizes the three comparisons; absolute MSE and Lift values appear in Appendix Tables 5 and 6. Paired state-bootstrap intervals appear in Appendix B.3. Because Lift uses the random mean as its denominator, absolute MSE differences and their intervals should also be considered. The main experiments use short response-prediction horizons to maintain comparable computational budgets across environment–model settings. Appendix C reports longer-horizon results and analyzes how mismatch changes with the prediction window.

4.2 Main Results

The gains of AWARE arise from candidate ranking. AWARE and the random control share the same pool of candidate action perturbations and differ only in selection. On CALVIN/Ctrl-World, AWARE outperforms the random control under all four action budgets, with Lift values of 46.80%, 54.09%, 68.87%, and 52.94%, respectively. On RoboCasa/Ctrl-World, AWARE likewise achieves positive gains under all four budgets, with Lift values of 99.9%, 98.7%, 79.9%, and 67.6%. Overall, AWARE maintains positive gains under small, medium, and large action budgets, although Lift does not increase monotonically with the budget. The shared candidate pool isolates the selection rule relative to uniform random choice. Across all 12 setting–budget comparisons, the 95% paired-bootstrap interval for the mean MSE difference excludes zero (Appendix Table 4).

Second world-model architecture. On CALVIN/IRA-SIM, AWARE exceeds the same-pool random control at all four budgets, with Lift values of 487.1%, 1473.1%, 389.4%, and 118.4% (Appendix Table 6).

Figure 4 shows the same-pool gains at four budgets for each dataset–world-model combination.

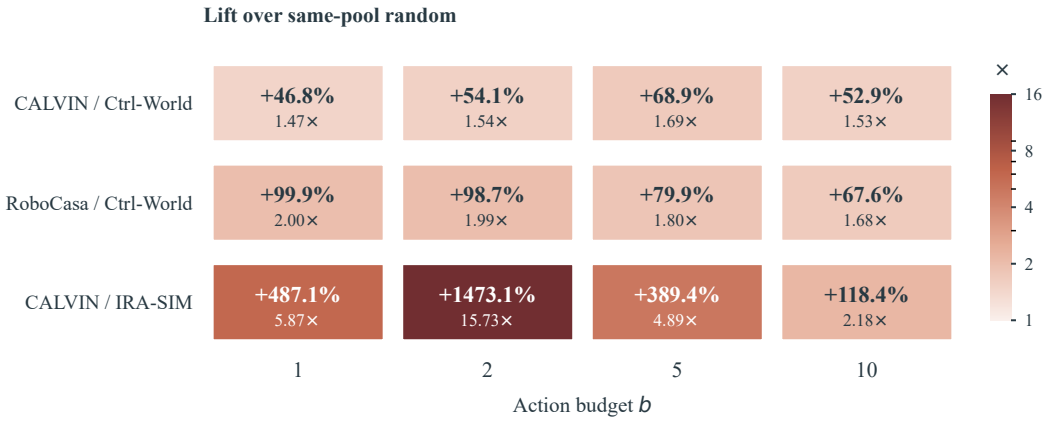


Figure 4: Same-pool mismatch gains at four action budgets. Rows show CALVIN/Ctrl-World, RoboCasa/Ctrl-World, and CALVIN/IRA-SIM. Cells report Lift and the selected-to-random MSE ratio; absolute values are in Appendix Tables 5 and 6.

4.3 Ablation Study

Relative action responses versus absolute responses. The effective ranking signal comes primarily from relative action responses rather than absolute responses. To test whether the gains of AWARE arise solely from selecting actions with large absolute model responses, we replace relative-response scoring with the absolute world-model response energy for each candidate. At $b = 1, 2, 5, 10$, the Lift values of the full AWARE relative to the random control are 46.80%, 54.09%, 68.87%, and 52.94%, respectively. With absolute-response scoring, the corresponding Lift values decrease to 6.31%, 5.46%, 12.74%, and 18.53%. Absolute-response scoring retains a weak positive signal but does not reproduce the substantial gains of the full AWARE across budgets. Absolute output magnitude alone is therefore insufficient to explain its effectiveness.

These scores encode different information. Let r_0 denote the predicted response under the nominal action and Δr_i the response increment induced by candidate perturbation i. The absolute response energy expands as

$$\|r_0 + \Delta r_i\|_2^2 = \|r_0\|_2^2 + 2\langle r_0, \Delta r_i\rangle + \|\Delta r_i\|_2^2. \tag{8}$$

The shared term $\|r_0\|_2^2$ does not affect within-pool ranking, but the cross term favors increments aligned with the nominal response and introduces state-dependent effects. By using $\|\Delta r_i\|_2^2$, AWARE isolates the model's marginal response to an action perturbation. The ablation supports relative directional responses, rather than absolute output magnitude, as the main source of selection gains (Appendix Table 8).

World-model response is not a direct differentiable surrogate for the final metric. To test world-model response energy as a direct optimization objective, Continuous AWARE replaces candidate generation and ranking with gradient ascent on this signal under the same L_2 perturbation budget. At $b = 1, 2, 5, 10$, optimized world-model response energy exceeds that of random directions by 70.47%, 24.06%, 59.45%, and 39.28%, respectively, confirming that Continuous AWARE optimizes the specified objective. However, its true action-response mismatch with the simulator exceeds the random control by only 17.84%, 4.36%, 1.48%, and 4.50%, respectively, substantially below the 46.80%, 54.09%, 68.87%, and 52.94% achieved by the full AWARE.

4.4 Post-attack Repair

We study post-attack repair as an exploratory experiment to test whether the difficult action examples selected by AWARE can improve a world model's action response; this does not establish general repair efficacy. All experiments start from the same Ctrl-World Base checkpoint fine-tuned on RoboCasa following the official procedure. Preliminary experiments found that simply adding examples while optimizing the diffusion loss did not consistently improve action-response fidelity. We therefore retain diffusion supervision while explicitly constraining response direction and magnitude relative to the simulator, RGB response error, and nominal-action prediction quality. Updates are restricted to the action encoder, selected spatial cross-attention projections, and a command-token branch. Appendix D gives the full objective and trainable scope.

Table 2: Effect of increasing repair states within episode 458. All checkpoints use the same eight evaluation anchors, distinct from the repair states, although temporal windows may overlap. Directional alignment is response cosine; magnitude error is absolute response-norm ratio error; adversarial prediction error is RGB MSE. Bold marks the best mean in each column.

Repair states	Directional alignment ↑	Magnitude error ↓	Adversarial prediction error ↓
Base	0.0397	0.9527	0.01596
1	0.0037	0.9477	0.01508
2	0.0131	0.8803	0.01412
4	**0.0761**	**0.7750**	**0.01291**

We first vary the number of repair states within episode 458. Table 2 shows that, as the count increases from one to four, directional alignment on the same eight evaluation anchors unused for repair rises from 0.0037 to 0.0761, magnitude error falls from 0.9477 to 0.7750, and adversarial prediction MSE falls from 0.01508 to 0.01291. All three means for the four-state checkpoint also improve over Base. This suggests that denser state coverage helps local within-episode repair.

Table 3: Transfer as the number of training episodes increases, with four repair states per episode. All checkpoints use the same eight episode-disjoint development episodes and two sampling draws per episode. Directional alignment is response cosine; magnitude error is absolute response-norm ratio error; response mismatch is raw RGB response RMSE; nominal and adversarial prediction errors are RGB MSE. Bold marks the best mean in each column.

Training episodes	Directional alignment ↑	Magnitude error ↓	Response mismatch ↓	Nominal prediction error ↓	Adversarial prediction error ↓
Base	0.0459	0.9308	**0.09833**	0.01419	0.01711
1	0.0546	**0.4533**	0.11001	0.01391	0.01762
2	0.0769	0.6388	0.10336	0.01310	0.01574
4	**0.1797**	0.5657	0.10222	**0.01253**	**0.01475**

We next fix four repair states per training episode and increase training-episode coverage. Table 3 shows that, from Base to the four-episode checkpoint, directional alignment on the fixed set of eight episode-disjoint development episodes rises from 0.0459 to 0.1797, magnitude error falls from 0.9308 to 0.5657, and nominal and adversarial prediction MSE fall from 0.01419 and 0.01711 to 0.01253 and 0.01475, respectively. Mean response RMSE, however, increases from 0.09833 to

0.10222. Thus, repair improves some component response and prediction metrics without improving mean pixel-level response mismatch on unseen episodes. Appendix D.2 analyzes this result.

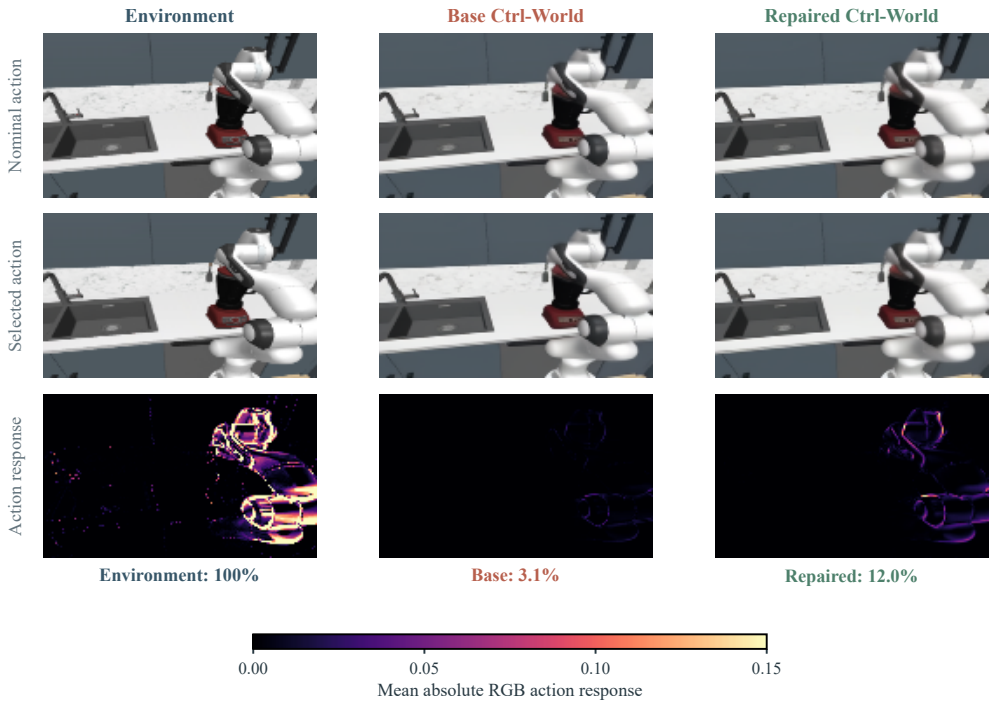


Figure 5: An under-response case from episode-disjoint development episode 318. Under the same nominal and AWARE-selected actions, we compare the environment, Base, and four-episode repair checkpoint. In the displayed frame, the norm of the model response relative to the environment increases from 3.1% for Base to 12.0% after repair. This illustrates local improvement; aggregate results on unseen episodes are in Table 3.

Overall, the small-scale repair experiment improves component measures of action response but does not demonstrate a reduction in mean response mismatch on unseen episodes. Additional repair visualizations, including a training-episode example, appear in Appendix D.1.

5 Conclusion

We introduce AWARE to identify weak spots at the commanded-action input of action-conditioned world models before deployment. With observation history and task semantics fixed, it ranks bounded candidates by their relative response in a frozen model; environment outcomes enter only offline evaluation. Local analysis identifies alignment conditions for this score and a limitation under direction-dependent under-response.

Across CALVIN/Ctrl-World, RoboCasa/Ctrl-World, and CALVIN/IRA-SIM, AWARE yields higher mean mismatch than same-pool random selection at all four budgets, with relative gains of 46.8–68.9%, 67.6–99.9%, and 118.4–1473.1%. Ablations suggest the value of relative responses and finite-pool ranking, although differing controls and search protocols preclude a matched component claim. Exploratory repair improves some response and prediction measures but not mean response RMSE on episode-disjoint data. Future work should test action clipping, multi-step control, and physical robots.

AI use statement

Generative AI tools assisted with Chinese translation, prose revision, and the mathematical decomposition and interpretation of the response-mismatch metric in the repair experiment. The authors are

responsible for the derivations, experimental results, citations, and final wording. These tools were not used to generate experimental data, implement the method, or run experiments.

Ethics statement

This work analyzes predictive mismatch under action perturbations in simulated robotic world models. The method may support pre-deployment diagnosis but could also be used to identify vulnerable actions. Its results should not be treated as recommendations for actions that are safe to execute on physical robots. Physical-robot safety was not validated; deployment requires task constraints, human review, and testing in the target environment.

Reproducibility statement

Section 3 specifies candidate construction, world-model response scoring, and selection; Section 3.4 gives the local analysis and its assumptions. Section 4.1 describes the experimental settings and metrics. Appendices B and D provide candidate distributions, model configurations, statistical units, ablation protocols, and repair details. Together, these specify the scope within which the reported results can be reproduced.

References

Anish Athalye, Logan Engstrom, Andrew Ilyas, and Kevin Kwok. Synthesizing robust adversarial examples. In *International Conference on Machine Learning*, 2018.

Jake Bruce, Michael D. Dennis, Ashley Edwards, Jack Parker-Holder, Yuge Shi, Edward Hughes, Matthew Lai, Aditi Mavalankar, Richie Steigerwald, Chris Apps, Yusuf Aytar, Sarah Bechtle, Feryal Behbahani, Stephanie C. Y. Chan, Nicolas Heess, Lucy Gonzalez, Simon Osindero, Sherjil Ozair, Scott Reed, Jingwei Zhang, Konrad Zolna, Jeff Clune, Nando de Freitas, Satinder Singh, and Tim Rocktäschel. Genie: Generative interactive environments. *arXiv preprint arXiv:2402.15391*, 2024.

Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *IEEE Symposium on Security and Privacy*, 2017.

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.

Jiayuan Gu, Fanbo Xiang, Xuanlin Li, Zhan Ling, Xiqiang Liu, Tongzhou Mu, Yihe Tang, Stone Tao, Xinyue Wei, Yunchao Yao, Xiaodi Yuan, Pengwei Xie, Zhiao Huang, Rui Chen, and Hao Su. ManiSkill2: A unified benchmark for generalizable manipulation skills. In *International Conference on Learning Representations*, 2023.

Yanjiang Guo, Lucy Xiaoyang Shi, Jianyu Chen, and Chelsea Finn. Ctrl-World: A controllable generative world model for robot manipulation. *arXiv preprint arXiv:2510.10125*, 2025.

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

Zhixiang Guo, Siyuan Liang, Shi Fu, Cheng Guo, Andras Balogh, Mark Jelasity, and Dacheng Tao. WMAttack: Automated attack search for adversarial evaluation of world-model agents. *arXiv preprint arXiv:2605.23220*, 2026b.

David Ha and Jürgen Schmidhuber. World models. *arXiv preprint arXiv:1803.10122*, 2018.

Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning latent dynamics for planning from pixels. In *International Conference on Machine Learning*, 2019.

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse domains through world models. *arXiv preprint arXiv:2301.04104*, 2023.

Nicklas Hansen and Xiaolong Wang. Hallucination in world models is predictable and preventable. *arXiv preprint arXiv:2606.27326*, 2026.

Anthony Hu, Lloyd Russell, Hudson Yeo, Zak Murez, George Fedoseev, Alex Kendall, Jamie Shotton, and Gianluca Corrado. GAIA-1: A generative world model for autonomous driving. *arXiv preprint arXiv:2309.17080*, 2023.

Sandy Huang, Nicolas Papernot, Ian Goodfellow, Yan Duan, and Pieter Abbeel. Adversarial attacks on neural network policies. *arXiv preprint arXiv:1702.02284*, 2017.

Andrew Ilyas, Logan Engstrom, Anish Athalye, and Jessy Lin. Black-box adversarial attacks with limited queries and information. In *International Conference on Machine Learning*, 2018.

Stephen James, Zicong Ma, David Rovick Arrojo, and Andrew J. Davison. RLBench: The robot learning benchmark and learning environment. *IEEE Robotics and Automation Letters*, 5(2): 3019–3026, 2020.

Feng Jiang, Yang Chen, Kyle Xu, Yuchen Liu, Haifeng Wang, Zhenhao Shen, Jasper Lu, Shengze Huang, Yuanfei Wang, Chen Xie, and Ruihai Wu. RoboWM-Bench: A benchmark for evaluating world models in robotic manipulation. *arXiv preprint arXiv:2604.19092*, 2026.

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

Hanqing Liu, Songping Wang, Jiahuan Long, Jiacheng Hou, Jialiang Sun, Chao Li, Yang Yang, Wei Peng, Xu Liu, Tingsong Jiang, Yao Mu, and Wen Yao. JailWAM: Jailbreaking world action models in robot control. *arXiv preprint arXiv:2604.05498*, 2026. doi: 10.48550/ARXIV.2604.05498. URL https://arxiv.org/abs/2604.05498.

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

Viktor Makoviychuk, Lukasz Wawrzyniak, Yunrong Guo, Michelle Lu, Kier Storey, Miles Macklin, David Hoeller, Nikita Rudin, Arthur Allshire, Ankur Handa, and Gavriel State. Isaac Gym: High performance GPU-based physics simulation for robot learning. *arXiv preprint arXiv:2108.10470*, 2021.

Oier Mees, Lukas Hermann, Erick Rosete-Beas, and Wolfram Burgard. CALVIN: A benchmark for language-conditioned policy learning for long-horizon robot manipulation tasks. *IEEE Robotics and Automation Letters*, 2022.

Soroush Nasiriany, Abhiram Maddukuri, Lance Zhang, Adeet Parikh, Aaron Lo, Abhishek Joshi, Ajay Mandlekar, and Yuke Zhu. RoboCasa: Large-scale simulation of everyday tasks for generalist robots. In *Robotics: Science and Systems*, 2024. URL https://www.roboticsproceedings.org/rss20/p050.html.

Lerrel Pinto, James Davidson, Rahul Sukthankar, and Abhinav Gupta. Robust adversarial reinforcement learning. In *International Conference on Machine Learning*, 2017.

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

Richard S. Sutton. Dyna, an integrated architecture for learning, planning, and reacting. *ACM SIGART Bulletin*, 2(4):160–163, 1991.

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.

Emanuel Todorov, Tom Erez, and Yuval Tassa. MuJoCo: A physics engine for model-based control. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5026–5033. IEEE, 2012.

Fanbo Xiang, Yuzhe Qin, Kaichun Mo, Yikuan Xia, Hao Zhu, Fangchen Liu, Minghua Liu, Hanxiao Jiang, Yifu Yuan, He Wang, Li Yi, Angel X. Chang, Leonidas J. Guibas, and Hao Su. SAPIEN: A simulated part-based interactive environment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020.

Sherry Yang, Yilun Du, Kamyar Ghasemipour, Jonathan Tompson, Leslie Kaelbling, Dale Schuurmans, and Pieter Abbeel. Learning interactive real-world simulators. *arXiv preprint arXiv:2310.06114*, 2023.

Tianhe Yu, Deirdre Quillen, Zhanpeng He, Ryan Julian, Avnish Narayan, Hayden Shively, Adithya Bellathur, Karol Hausman, Chelsea Finn, and Sergey Levine. Meta-World: A benchmark and evaluation for multi-task and meta reinforcement learning. In *Conference on Robot Learning*, 2020.

Siyuan Zhou, Yilun Du, Jiaben Chen, Yandong Li, Dit-Yan Yeung, and Chuang Gan. RoboDreamer: Learning compositional world models for robot imagination. *arXiv preprint arXiv:2404.12377*, 2024.

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

A Theoretical Analysis and Proofs

Fix a state s and nominal action a. For the same-representation case in the following regime proofs, let $J_W, J_E \in \mathbb{R}^{D\times d}$ denote the model and environment action-response Jacobians. The covariance proof below also holds when the score and error use different representations, as defined in Eq. (A.2). For $d \geq 2$ and $b > 0$, define the local quadratic approximations

$$\begin{aligned} A &= J_W^\top J_W, & q(\delta) &= \delta^\top A\delta, \\ B &= (J_W - J_E)^\top (J_W - J_E), & m(\delta) &= \delta^\top B\delta. \end{aligned} \tag{A.1}$$

Here, q and m approximate world-model response energy and action-response mismatch, respectively. We omit positive element-count normalizations, which leave each score's ranking unchanged. If J_Q is the Jacobian of the representation actually scored, J_x that of decoded RGB, and J_E that of the aligned environment response, the corresponding general pair is

$$A_Q = J_Q^\top J_Q, \qquad B_x = (J_x - J_E)^\top (J_x - J_E), \tag{A.2}$$

with $q(\delta) = \delta^\top A_Q\delta$ and $m(\delta) = \delta^\top B_x\delta$. Same-space scoring has $J_Q = J_x = J_W$; latent scoring has $J_Q = J_z$ and $J_x = GJ_z$ for decoder Jacobian G. Restoring a shared normalization D^{-1} multiplies covariances and variances by D^{-2}. For a symmetric matrix H, write its anisotropic component as $H_0 = H - \mathrm{tr}(H)I/d$.

A.1 Continuous Optimization and Finite Candidate Selection

Continuous optima. The Euclidean gradient is $\nabla q = 2A\delta$. On $\|\delta\|_2 = b$,

$$\nabla_{\mathbb{S}} q = 2\left(A\delta - \frac{q(\delta)}{b^2}\delta\right), \qquad \max_{\|\delta\|_2=b} q(\delta) = b^2\lambda_{\max}(A). \tag{A.3}$$

Replacing A with B gives the corresponding results for m. The last equality follows from the Rayleigh–Ritz theorem: the set of maximizers is the intersection of the leading eigenspace and the radius-b sphere. The objectives therefore share a maximizer if and only if their leading eigenspaces have a nonzero intersection. At a fixed radius, $B = cA + \gamma I$, $c > 0$, is sufficient to preserve the entire ranking, since $m = cq + \gamma b^2$. Positive average correlation does not guarantee this property.

Counterexample. Consider $A = \mathrm{diag}(3, 8/3, 0)$ and $B = \mathrm{diag}(0, 3, 0)$. Although $\langle A_0, B_0 \rangle_F = 7/3 > 0$, q is maximized at $\pm be_1$, where m is zero, below the random-direction mean $b^2 \,\mathrm{tr}(B)/3 = b^2$. This construction is realized by $J_W = \mathrm{diag}(\sqrt{3}, \sqrt{8/3}, 0)$ and $J_E = \mathrm{diag}(\sqrt{3}, \sqrt{8/3} + \sqrt{3}, 0)$. Thus, successfully optimizing world-model response need not increase true mismatch.

Proposition A.1 (Expected gain from finite candidate selection)**.** *Let $K \geq 1$ and let $\Delta_1, \ldots, \Delta_K$ be independent draws from any common distribution of executable action perturbations at a fixed state. Set $q_i = q(\Delta_i)$ and $m_i = m(\Delta_i)$ for model scores and integrable evaluation discrepancies. Let $F(t) = \Pr(q(\Delta) \leq t)$ and $F(t^-) = \Pr(q(\Delta) < t)$. If I is the smallest index attaining $\max_i q_i$, define $w_K(t) = \sum_{j=0}^{K-1} F(t^-)^j F(t)^{K-1-j}$. Then*

$$\mathbb{E}[m_I] - \mathbb{E}[m(\Delta)] = \mathrm{Cov}(m(\Delta), w_K(q(\Delta)))\,. \tag{A.4}$$

Proof. Conditional on candidate i having score t, it wins precisely when the $i-1$ earlier scores are strictly below t and the $K-i$ later scores are at most t. Independence gives probability $F(t^-)^{i-1} F(t)^{K-i}$. Summing over indices yields $\mathbb{E}[m_I] = \mathbb{E}[m(\Delta) w_K(q(\Delta))]$. Applying the same argument with $m \equiv 1$ shows $\mathbb{E}[w_K(q(\Delta))] = 1$; subtracting the mean discrepancy gives Eq. (A.4). The unconditional expectation of a uniformly selected same-pool candidate is exactly $\mathbb{E}[m(\Delta)]$. This identity permits variable radii, projection or clipping, different score/error representations, and a non-time-aligned target; it does not imply a positive gain. □

If $g(t) = \mathbb{E}[m(\Delta) \mid q(\Delta) = t]$ is nondecreasing, Eq. (A.4) is nonnegative because $w_K(t)$ is nondecreasing. This follows from the covariance identity for independent identically distributed Q, Q':

$$2\,\mathrm{Cov}(g(Q), w_K(Q)) = \mathbb{E}[(g(Q) - g(Q'))(w_K(Q) - w_K(Q'))]. \tag{A.5}$$

For continuous score distributions, $F(t^-) = F(t)$ almost surely and $w_K(t) = K F(t)^{K-1}$, recovering the usual order-statistic expression. The condition concerns expected gain under the actual candidate distribution and evaluation metric. Positive $\mathrm{Cov}(q, m)$ alone cannot replace it or guarantee improvement in every realized pool. The formula includes clipped-action score ties through the stated index rule, but neither establishes the required monotonic relation nor attributes measured Lift to directional geometry.

A.2 Response-Mismatch Regimes and Information Limits

Uniform scaling. In the ideal over-response regime $J_E = \varepsilon J_W$, $0 \leq \varepsilon < 1$, we have $B = (1-\varepsilon)^2 A$. In the uniform under-response regime $J_W = \alpha J_E$, $0 < \alpha < 1$, $A = \alpha^2 S$ and $B = (1-\alpha)^2 S$, where $S = J_E^\top J_E$.

In both cases, m is a positive rescaling of q, so the rankings coincide. When $A_0 = 0$, all fixed-radius directions receive the same score: ranking agreement then provides no nontrivial directional information. When $J_W = 0$, the model response likewise cannot distinguish directions.

Nonuniform under-response. With $J_W = J_E R$ and $S = J_E^\top J_E$, the same-space score and mismatch geometries are

$$A = R^\top S R, \qquad B = (R - I)^\top S (R - I). \tag{A.6}$$

This representation alone does not ensure under-response in every direction. The necessary and sufficient condition is

$$\|J_W u\|_2 \leq \|J_E u\|_2 \quad (\forall u) \quad \Longleftrightarrow \quad R^\top S R \preceq S. \tag{A.7}$$

If S and R are symmetric and share an orthonormal eigenbasis V, with eigenvalues $s_j \geq 0$ and $0 \leq r_j \leq 1$, then

$$A = V\,\mathrm{diag}(s_j r_j^2) V^\top, \qquad B = V\,\mathrm{diag}(s_j (1 - r_j)^2) V^\top. \tag{A.8}$$

Direction-dependent attenuation can reverse the ranking. For example, $J_E = I$ and $J_W = R = \mathrm{diag}(3/4, 1/4)$ satisfy Eq. (A.7), but yield $A = \mathrm{diag}(9/16, 1/16)$ and $B = \mathrm{diag}(1/16, 9/16)$, which favor opposite directions. Without a shared eigenbasis, one must use Eq. (A.7) and the full matrices A, B, rather than compare coordinates individually.

Proposition A.2 (Non-identifiability of the optimal selection under an unknown environment)**.** *A selector that observes only the world model and candidate actions cannot guarantee selection of the maximum-mismatch candidate for every direction-dependent under-response environment.*

Proof. Consider exact linear responses, the candidate set $\{be_1, be_2\}$, and a shared $J_W = \mathrm{diag}(2, 1)$. Two possible environments and their mismatch geometries are

$$\begin{aligned} J_E^{(1)} &= \mathrm{diag}(4, 11/10), \quad B^{(1)} = \mathrm{diag}(4, 1/100), \\ J_E^{(2)} &= \mathrm{diag}(21/10, 4), \quad B^{(2)} = \mathrm{diag}(1/100, 9). \end{aligned} \tag{A.9}$$

In both environments, response magnitudes are at least as large as the model response magnitude in every direction.

The two environments have different optimal candidates, despite the identical observed model and candidate set. If the selector chooses be_1 with probability p, its success probabilities are p and $1 - p$ in the two environments. Its worst-case success probability is therefore at most $1/2$. □

This result characterizes the information limit without environment feedback. Effective selection requires an additional response-alignment property; under-response in every direction is not sufficient by itself. For a latent score with decoder Jacobian G, the matrices are $A = J_z^\top J_z$ and $B = (GJ_z - J_E)^\top (GJ_z - J_E)$, not the same-space pair above. For example, $J_z = \mathrm{diag}(2, 1)$, $G = \mathrm{diag}(0.1, 10)$, and $J_E = 0$ give latent scores $(4, 1)$ but RGB squared responses $(0.04, 100)$ on the two equal-radius coordinate directions. Thus even zero environment response does not guarantee latent-score/RGB-error ranking agreement; this illustrative construction is not a measurement of the tested models.

A.3 Geometric Alignment and the Validity of Local Approximations

Spherical covariance. For $U \sim \mathrm{Unif}(\mathbb{S}^{d-1})$, rotational symmetry and $\|U\|_2 = 1$ give

$$\mathbb{E}[U_i U_j] = \frac{\delta_{ij}}{d}, \qquad \mathbb{E}[U_i U_j U_k U_l] = \frac{\delta_{ij}\delta_{kl} + \delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk}}{d(d+2)}, \tag{A.10}$$

Here, δ_{ij} is the Kronecker delta. Letting $Q = q(bU)$ and $M = m(bU)$, we have $\mathbb{E}[Q] = b^2 \,\mathrm{tr}(A)/d$ and $\mathbb{E}[M] = b^2 \,\mathrm{tr}(B)/d$. Coordinate expansion yields

$$\mathbb{E}[QM] = \frac{b^4}{d(d+2)} \big[\mathrm{tr}(A)\,\mathrm{tr}(B) + 2\,\mathrm{tr}(AB)\big]. \tag{A.11}$$

Subtracting the product of means and using $\mathrm{tr}(AB) - \mathrm{tr}(A)\,\mathrm{tr}(B)/d = \langle A_0, B_0 \rangle_F$ gives

$$\mathrm{Cov}\big(q(bU), m(bU)\big) = \frac{2b^4}{d(d+2)} \langle A_0, B_0 \rangle_F. \tag{A.12}$$

The identity also holds for the general pair in Eq. (A.2); its same-space regime interpretations do not explain the excluded historical RoboCasa statistic with a temporally unaligned reference.

Independent variable radii. For $\Delta = RU$, with nonnegative radius R independent of uniform direction U, the law of total covariance gives an exact decomposition for the local quadratic scores:

$$\mathrm{Cov}\big(q(\Delta), m(\Delta)\big) = \frac{2\mathbb{E}[R^4]}{d(d+2)} \langle A_0, B_0 \rangle_F + \frac{\mathrm{Var}(R^2)}{d^2}\,\mathrm{tr}(A)\,\mathrm{tr}(B). \tag{A.13}$$

For positive-semidefinite A, B, the second term is nonnegative and arises from radius variation even if directional alignment vanishes. This decomposition is not an empirical attribution of the measured Lift.

The independent radial/spherical assumption does not describe post-projection or clipped executable perturbations; the general finite-pool identity above still applies under its independent-draw and tie-breaking assumptions.

Setting the two matrices equal gives the variances. For $A_0, B_0 \neq 0$,

$$\mathrm{Corr}(Q, M) = \frac{\langle A_0, B_0 \rangle_F}{\|A_0\|_F \|B_0\|_F}. \tag{A.14}$$

Normalized Frobenius alignment therefore equals the Pearson correlation of the local quadratic scores; it does not measure rank correlation or agreement between optimal directions. If either centered matrix is zero, the corresponding score is constant on the fixed-radius sphere and the correlation coefficient is undefined.

A sufficient condition under over-response. Write $B = A + C$, where $C = -J_W^\top J_E - J_E^\top J_W + J_E^\top J_E$. Cauchy–Schwarz gives

$$\langle A_0, B_0 \rangle_F \geq \|A_0\|_F (\|A_0\|_F - \|C_0\|_F). \tag{A.15}$$

Thus, $\|C_0\|_F < \|A_0\|_F$ guarantees strictly positive alignment. Because centering is an orthogonal projection,

$$\|C_0\|_F \leq \|C\|_F \leq 2\|J_W\|_2 \|J_E\|_F + \|J_E\|_F^2, \tag{A.16}$$

Requiring the right-hand side to be smaller than $\|A_0\|_F$ yields a more conservative sufficient condition. A relatively small environment response is not enough: the cross terms must be small relative to the anisotropic component of the model response. This is a theoretical condition, not a directly observable pre-deployment selection score.

Approximation error. Suppose the Jacobians of $f_W(a) = x_W(s, a)$ and $f_E(a) = x_E(s, a)$ are Lipschitz in operator norm, with constants L_W, L_E, in a neighborhood containing $\{a + t\delta : 0 \leq t \leq 1\}$. The Taylor remainder satisfies $\|r_W(\delta)\|_2 \leq L_W \|\delta\|_2^2/2$, with an analogous bound for the environment. Expanding the squared norms and applying Cauchy–Schwarz, for $\|\delta\|_2 = b$,

$$\begin{aligned} \left| \|\Delta x_W\|_2^2 - q(\delta) \right| &\leq \|J_W\|_2 L_W b^3 + \frac{L_W^2 b^4}{4}, \\ \left| \|\Delta x_W - \Delta x_E\|_2^2 - m(\delta) \right| &\leq \|J_W - J_E\|_2 (L_W + L_E) b^3 + \frac{(L_W + L_E)^2 b^4}{4}. \end{aligned} \tag{A.17}$$

These are absolute error bounds. Dominance of the quadratic term additionally requires a nondegenerate quadratic contribution and a sufficiently small perturbation. Large perturbations can leave the smooth neighborhood, and contact switches can violate smoothness. After action projection, the bounds must use the executed perturbation; the spherical covariance result still requires fixed-radius sampling. Finite-budget experiments therefore require independent validation: local correlation does not imply a global selection guarantee.

B Experimental Implementation and Reproducibility Details

This appendix supplements Section 4 with details of candidate selection, environment–model configurations, response metrics, and ablation implementations. Attack selection uses only the frozen world model; environment responses are used for offline evaluation after selection.

B.1 Candidate Selection and Information Access

Algorithm B.1 gives the complete selection procedure. The executable-action map $\Pi_{\mathcal{A}}$ enforces environment-specific action constraints and is the identity when no projection is used. Each candidate

is mapped to $a_i = \Pi_{\mathcal{A}}(a + \widetilde{\delta}_i)$, with executed perturbation $\delta_i = a_i - a$. Both model predictions and subsequent environment evaluations use this executable action.

The response representation is fixed within each experiment and can be decoded video or generated latents. Its dimensionality is D. After caching the nominal prediction, candidates are scored by the relative response energy in Eq. (2). A fixed index order resolves ties. The same-pool random control samples uniformly from the returned pool P; environment responses are queried only for offline evaluation after selection.

Algorithm B.1 Pre-deployment candidate selection

Input: Frozen model W; state or observation s; nominal action a; budget b; candidate count K; executable-action map $\Pi_{\mathcal{A}}$; fixed conditioning c and generation seed ξ; response dimension D
Output: Selected action a^*, executed perturbation δ^*, and scored pool P

Sample K perturbations $\{\widetilde{\delta}_i\}_{i=1}^{K}$ under budget b (Appendix B.2).
Cache the nominal prediction $x_0 \leftarrow W(s, a; c, \xi)$.
for $i = 1, \ldots, K$ **do**
    Map to an executable action $a_i \leftarrow \Pi_{\mathcal{A}}(a + \widetilde{\delta}_i)$.
    Record the executed perturbation $\delta_i \leftarrow a_i - a$.
    Predict $x_i \leftarrow W(s, a_i; c, \xi)$ with the same conditioning and seed.
    Compute the relative-response score $q_i \leftarrow \|x_i - x_0\|_2^2 / D$.
end for
Collect the indexed pool $P \leftarrow \big((a_i, \delta_i, q_i)\big)_{i=1}^{K}$.
Select $i^* \in \arg\max_i q_i$, breaking ties by the smallest index.
return a_{i^*}, δ_{i^*}, P

Scoring the full pool requires one nominal prediction and K candidate predictions. These counts refer to prediction instances; the number of batched calls depends on the implementation. Nominal and candidate predictions share the checkpoint, context, text conditions, and diffusion seed. Simulator futures, task-success labels, and final mismatch are excluded from candidate generation, scoring, and selection.

B.2 Candidate Distributions and Model Configurations

Section 3.4 analyzes local directional geometry on a fixed-radius sphere. In the discrete experiments, b is an upper bound on the seven-dimensional normalized L_2 perturbation, rather than a physical distance or percentage of actuator range, and candidate pools use variable radii. Projection onto executable actions can further reduce the perturbation norm. Thus, the theory compares directions at a given radius, whereas the experiments vary both direction and magnitude. Within each main-table paired comparison, AWARE and the random control share the candidate pool and clipping rule. RoboCasa executable actions have clipping records at each budget; large-budget gains also include boundary-action effects. The analyzed CALVIN archive has no corresponding clipping-trigger flags.

All three environment–model settings contain 90 evaluation states per budget and perturb only seven components of the first future action. Their candidate distributions and response representations are described below.

CALVIN / Ctrl-World. For each $b \in \{1, 2, 5, 10\}$, we construct 64 variable-strength candidates by normalizing a seven-dimensional Gaussian vector to obtain a direction and independently sampling a radius uniformly from $[0.05, b]$. The perturbation acts on the relative end-effector action, including the gripper coordinate. Only the first action of the five-action conditioning segment is replaced; the remaining four remain nominal. Scoring uses the relative visual MSE over five decoded frames at a fixed evaluation view and resolution 192×200.

RoboCasa / Ctrl-World. The main task is `CloseBlenderLid`. We construct the candidate action set using the same settings as for CALVIN/Ctrl-World. The perturbation acts on simulator action dimensions `[5:12]`; the other five dimensions remain unchanged.

CALVIN / IRA-SIM. The cross-model experiment reuses the CALVIN states, action manifests, and 64-element pools, replacing only the selector's world model and response representation with

IRA-SIM. The model is trained from scratch for 70k steps on the converted CALVIN-D subset, without Ctrl-World initialization. Candidates are reranked by IRA-SIM latent response energy, and environment references are taken from the same CALVIN replay archive.

B.3 Response Metrics and Statistical Units

For a response tensor with T frames, spatial dimensions $H \times W$, and C channels, the reported action-response mismatch is the elementwise mean

$$M(s,a,\delta) = \frac{1}{THWC} \sum_{t,h,w,c} \left[\left(x_E^{\text{adv}} - x_E^{\text{nom}}\right)_{t,h,w,c} - \left(x_W^{\text{adv}} - x_W^{\text{nom}}\right)_{t,h,w,c} \right]^2. \tag{B.1}$$

This metric compares changes in model and environment responses to perturbed versus nominal actions at the same state. It differs from prediction MSE, which compares a single predicted video with an environment video.

Comparisons are paired at the state level. Each main experiment compares selected mismatch against the mean over eight distinct same-pool random actions per state. Lift is computed from the ratio of unrounded means in Eq. (7), rather than averaging state-level ratios. We resample the 90 paired states with replacement for 20,000 replicates, retaining each state's eight controls together, and report percentile 95% intervals. Resampling uses a fixed local pseudorandom seed and does not rerun the model or simulator. Table 4 reports the main-experiment Lift intervals.

Table 4: Paired state-bootstrap 95% percentile intervals for Lift (%) in the main experiments. Each cell uses 90 paired states and 20,000 state resamples, retaining the eight random controls for each state together. Point estimates appear in Tables 5 and 6.

Setting	$b = 1$	$b = 2$	$b = 5$	$b = 10$
CALVIN / Ctrl-World	31.1–64.9	35.1–77.4	47.6–95.0	38.0–71.1
RoboCasa / Ctrl-World	84.2–116.5	82.7–116.6	66.1–94.7	54.5–81.8
CALVIN / IRA-SIM	427.3–556.8	1186.0–1850.5	328.8–467.1	105.1–133.4

B.4 Detailed Main-Experiment Results

The tables below report the absolute mismatch MSE and Lift summarized in Figure 4. They use the same-pool controls and budgets described in Section 4.1.

Table 5: AWARE yields higher action-response mismatch than the same-pool random baseline across two environments with the same world model.

Metric	Method	$b = 1$	$b = 2$	$b = 5$	$b = 10$
Panel A: CALVIN / Ctrl-World					
Action-response mismatch MSE ↑	**AWARE**	$\mathbf{1.78 \times 10^{-4}}$	$\mathbf{3.26 \times 10^{-4}}$	$\mathbf{6.30 \times 10^{-4}}$	$\mathbf{8.58 \times 10^{-4}}$
	Random	1.21×10^{-4}	2.12×10^{-4}	3.73×10^{-4}	5.61×10^{-4}
Lift (%) ↑		**46.80**	**54.09**	**68.87**	**52.94**
Panel B: RoboCasa / Ctrl-World					
Action-response mismatch MSE ↑	**AWARE**	$\mathbf{6.99 \times 10^{-3}}$	$\mathbf{8.74 \times 10^{-3}}$	$\mathbf{9.80 \times 10^{-3}}$	$\mathbf{9.98 \times 10^{-3}}$
	Random	3.49×10^{-3}	4.40×10^{-3}	5.45×10^{-3}	5.95×10^{-3}
Lift (%) ↑		**99.9**	**98.7**	**79.9**	**67.6**

B.5 Ablation Configurations

Absolute-response scoring. The intended change replaces Eq. (2) with the absolute energy of the perturbed model output. State-level records reveal a difference in the control denominator: the absolute-response run evaluates one random candidate per state, whereas the full method averages eight. Both use the same source candidate pools, but their reported random means differ; at $b = 1$

Table 6: Cross-model results on CALVIN/IRA-SIM.

Metric	Method	$b = 1$	$b = 2$	$b = 5$	$b = 10$
Action-response mismatch MSE ↑	**AWARE**	**0.001177**	**0.013778**	**0.038112**	**0.039403**
	Random	0.000201	0.000876	0.007788	0.018042
Lift (%) ↑		**487.1**	**1473.1**	**389.4**	**118.4**

they are 7.0326×10^{-5} and 1.2131×10^{-4}, respectively. Hence, the cross-run Lift difference is not a paired estimate of changing only the score. Intervals for the absolute-response run itself appear in Table 7.

Continuous optimization. Continuous AWARE differentiates decoded Ctrl-World relative-response MSE with respect to the seven-dimensional perturbation. It uses Adam at learning rate 0.03, ten updates, and two random restarts. After each update, the perturbation is projected onto the radius-b L_2 sphere, and the best iterate by model response is retained. The random control uses eight exactly equal-norm directions; the model seed is 1234. Neither simulator responses nor final mismatch are available during optimization. This ablation tests whether continuously increasing model response yields true-mismatch gains; its search procedure and query cost differ from finite candidate ranking.

Table 7: Paired state-bootstrap 95% percentile intervals for ablation Lift (%). The absolute-response run uses one random control per state; the continuous run uses eight fixed-norm random directions per state. Each interval resamples 90 paired states 20,000 times and pertains only to its own protocol.

Protocol	$b = 1$	$b = 2$	$b = 5$	$b = 10$
Absolute-response scoring	−5.5–19.5	−10.8–24.6	−6.0–35.6	2.7–36.7
Continuous response optimization	9.1–26.8	−3.1–12.0	−6.3–9.5	−2.8–12.4

B.6 Detailed Ablation Results

Table 8 reports the absolute mismatch MSE and the random reference used within each ablation protocol. The relative-response and continuous-optimization runs use different controls, as described in Section B.5.

Table 8: Ablation results on CALVIN/Ctrl-World.

Method	Metric	$b = 1$	$b = 2$	$b = 5$	$b = 10$
Full AWARE	Action-response mismatch MSE ↑	1.7808×10^{-4}	3.2622×10^{-4}	6.2956×10^{-4}	8.5768×10^{-4}
	Random MSE	1.2131×10^{-4}	2.1170×10^{-4}	3.7282×10^{-4}	5.6081×10^{-4}
	Lift (%) ↑	**46.80**	**54.09**	**68.87**	**52.94**
AWARE w/o relative energy	Action-response mismatch MSE ↑	7.4762×10^{-5}	1.3190×10^{-4}	2.8171×10^{-4}	4.9056×10^{-4}
	Random MSE	7.0326×10^{-5}	1.2507×10^{-4}	2.4988×10^{-4}	4.1387×10^{-4}
	Lift (%) ↑	6.31	5.46	12.74	18.53
AWARE w/o candidate search	Action-response mismatch MSE ↑	1.8011×10^{-4}	2.7669×10^{-4}	5.0788×10^{-4}	6.9156×10^{-4}
	Random MSE	1.5284×10^{-4}	2.6512×10^{-4}	5.0048×10^{-4}	6.6177×10^{-4}
	Lift (%) ↑	17.84	4.36	1.48	4.50

C Error Accumulation in Long-Horizon Predictions

We evaluate action-response mismatch over five frames to isolate the local response to a single action intervention, with observation history and task conditions fixed. Longer windows combine

initial response errors with subsequent state transitions and recursive prediction errors, increasingly measuring the stability of the full prediction process. The following decomposition and experiments explain this distinction.

C.1 Sources of Long-Horizon Mismatch

Nominal and perturbed trajectories share the initial state and all recorded actions after the first intervention. Denote the model and environment trajectories by $x^0_{W,t}, x^+_{W,t}$ and $x^0_{E,t}, x^+_{E,t}$, respectively. With prediction errors $e^0_t = x^0_{W,t} - x^0_{E,t}$ and $e^+_t = x^+_{W,t} - x^+_{E,t}$, per-frame mismatch satisfies

$$\begin{aligned} M_t &= \frac{1}{D}\left\|(x^+_{W,t} - x^0_{W,t}) - (x^+_{E,t} - x^0_{E,t})\right\|_2^2 \\ &= \frac{1}{D}\|e^+_t - e^0_t\|_2^2 = \frac{\|e^+_t\|_2^2 + \|e^0_t\|_2^2 - 2\langle e^+_t, e^0_t\rangle}{D}. \end{aligned} \tag{C.1}$$

Relative responses cancel errors shared by both branches. During recursive rollout, however, generated states condition subsequent predictions: intervention-induced state differences affect later prediction errors, while environment differences propagate through dynamics and contacts. Even with identical subsequent actions, M_t therefore depends on both branch-error magnitudes and their inner product. Error accumulation can increase mismatch, whereas cancellation can decrease it.

Thus, a larger long-window mean $T^{-1}\sum_{t=1}^{T} M_t$ need not indicate a larger direct response error to the initial action; a smaller mean need not indicate correction. This measure of overall prediction consistency after intervention cannot directly replace our local response metric.

C.2 Supplementary Evidence from Recursive Prediction

We examine six `CloseBlenderLid` cases, selecting one action per case from 128 candidates at $b = 10$ by the frozen Ctrl-World model's relative latent response. Only the first low-level control step is perturbed; both branches execute identical recorded actions for the next 159 steps. The model recursively generates ten blocks, conditioning each on the preceding block's final latent frame, yielding 41 frames over 0–8 seconds. Both branches share the block-wise seed policy with base seed 1234.

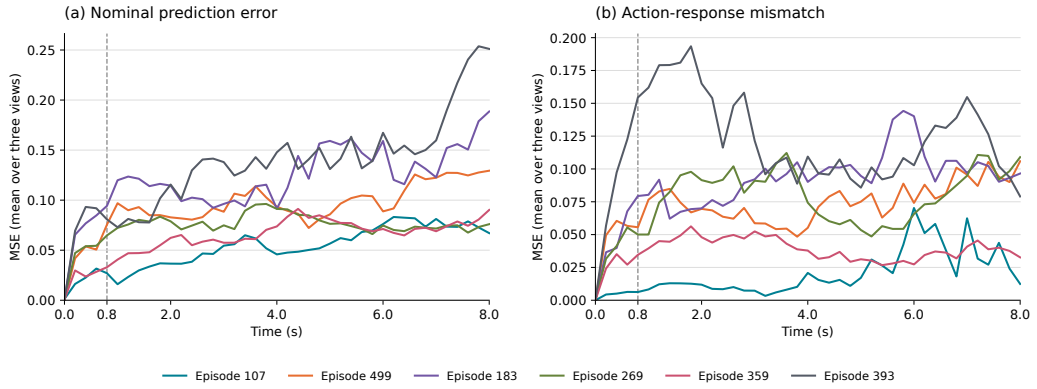


Figure C.1: Recursive-rollout nominal prediction MSE (left) and response mismatch (right; Eq. (C.1)). Curves are unsmoothed left/right/wrist-view means for six individual cases; vertical scales differ. The dashed line ends the first block (0.8 seconds), which includes the pre-action frame; its indices differ from the main five-frame window.

At 0.8, 4, and 8 seconds, mean nominal prediction MSE is 0.06254, 0.09058, and 0.13372, respectively; mean mismatch is 0.06339, 0.06462, and 0.07258. The two quantities do not grow in tandem, and individual curves fluctuate substantially (Figure C.1). Consistent with Eq. (C.1), mismatch depends on both branches' evolving errors, rather than one trajectory's prediction quality or prediction duration alone.

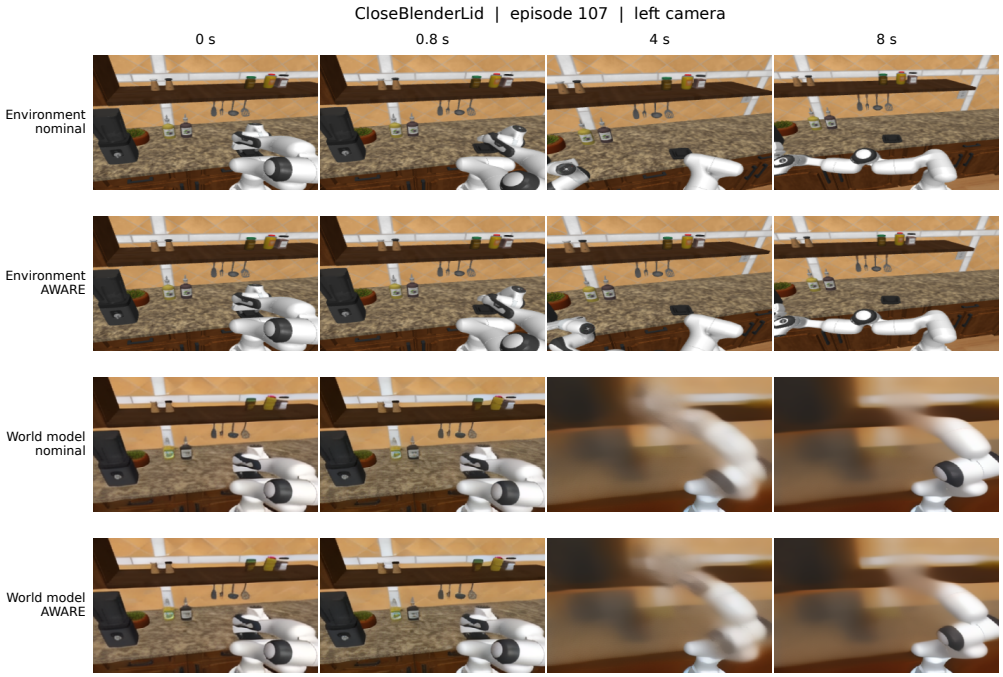


Figure C.2: Episode 107 (first manifest case, left view) at $0, 0.8, 4, 8$ seconds. Rows: nominal/attacked environment, then nominal/attacked model. Both predictions develop blur and positional errors. Attack attribution requires both nominal references and relative responses, beyond comparing the attacked prediction with the environment.

These results illustrate how model drift and post-intervention error propagation complicate longer-window mismatch. They provide diagnostic support for the chosen window, not a systematic rollout-length ablation, and do not re-estimate the main-table attack gains.

D Repair Experiment Details

Repair data and trainable scope. Every scale setting is initialized from the same unrepaired Ctrl-World Base checkpoint. For each repair state, we follow the AWARE procedure to construct 128 candidate actions in the native seven-dimensional action space under budget $b = 1$. The frozen model selects a hard action according to its relative-response score before simulator labels are queried. Clean and attacked executions are paired from the same initial state. Supervision comprises the current frame and four subsequent frames from three camera views, conditioned on the intervening action sequence.

During repair, we update layers 0, 2, and 4 of the action encoder, the value and output projections of spatial cross-attention, and a command-token branch initialized to produce zero output. Temporal Transformer blocks remain frozen.

Objective and optimization. Let $\Delta y = y^{\text{adv}} - y^{\text{nom}}$ denote the RGB action response observed in the environment and $\Delta\hat{y} = \hat{y}^{\text{adv}} - \hat{y}^{\text{nom}}$ denote the model's response. Both concatenate the four future frames across the three views. We optimize

$$\begin{aligned}\mathcal{L} = \mathcal{L}_{\text{RGB-response}} + \mathcal{L}_{\text{RGB-nominal}} + \mathcal{L}_{\text{direction}} + \tfrac{1}{2}\mathcal{L}_{\text{magnitude}} \\ + \tfrac{1}{4}\big(\mathcal{L}_{\text{latent-response}} + \mathcal{L}_{\text{latent-nominal}}\big) + \tfrac{1}{2}\big(\mathcal{L}_{\text{diff-clean}} + \mathcal{L}_{\text{diff-attacked}}\big).\end{aligned} \tag{D.1}$$

The direction loss is $1 - \cos(\Delta\hat{y}, \Delta y)$, and the magnitude loss is $(\|\Delta\hat{y}\|_2/\|\Delta y\|_2 - 1)^2$. The RGB-response and RGB-nominal terms use squared error. The two diffusion terms retain factual supervision under clean and attacked conditions. AdamW uses learning rates of 10^{-5} for native

parameters and 10^{-3} for token parameters, zero weight decay, and global gradient clipping at 1.0. Paired sampling and factual random seeds are fixed across training runs.

Evaluation. The state-count experiment uses the same eight evaluation anchors $\{100, 116, 148, 164, 204, 224, 240, 260\}$ in episode 458. These anchors are excluded from the repair-state sets, so this experiment measures the effect at other states within the same episode. The episode-count experiment fixes four repair states per training episode and evaluates on the same episode-disjoint development set $\{470, 477, 127, 342, 179, 318, 240, 494\}$, with two sampling draws per episode.

D.1 Supplementary Repair Visualizations

The bar plots visualize the aggregate repair measurements reported in Tables 2 and 3. The training-episode example illustrates a local response recovery and is separate from the episode-disjoint development results.

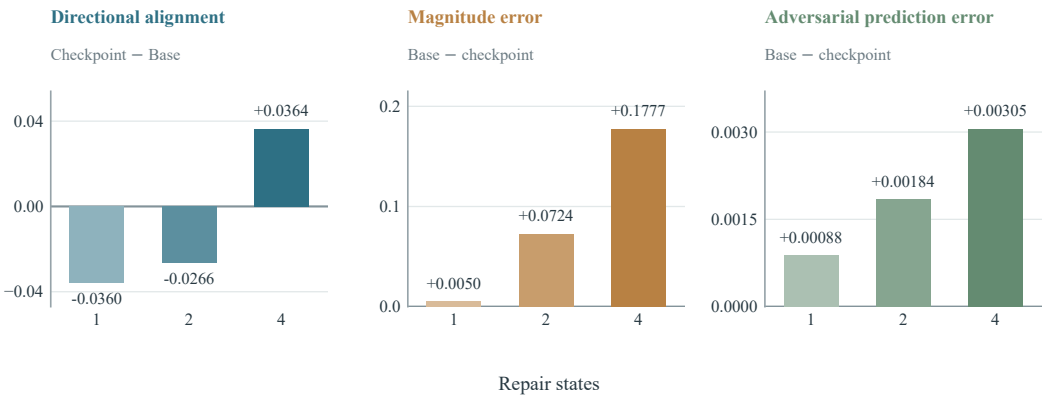


Figure D.1: Changes relative to Base as the number of repair states increases within episode 458; metrics are reported in Table 2.

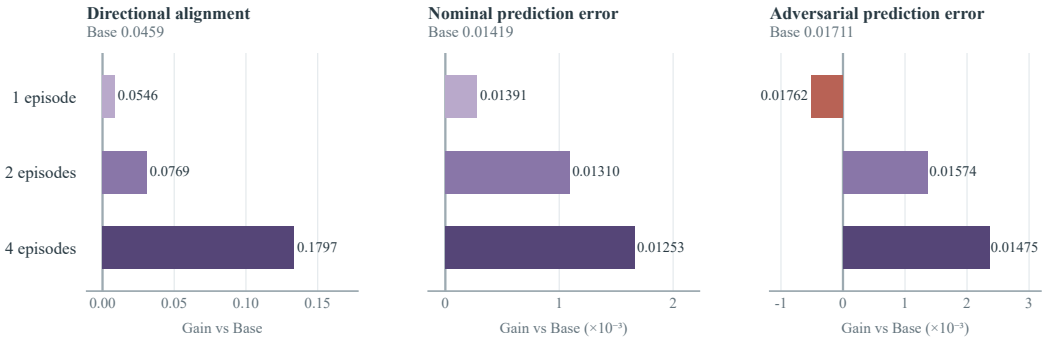


Figure D.2: Trends as the number of training episodes increases, with four repair states per episode.

D.2 Relationships among Response Magnitude, Direction, and Mismatch

The directional cosine, norm-ratio error, nominal and attacked prediction MSE, and response RMSE in Section 4.4 are distinct functions of the same predictions. The following identities explain how their means can move in different directions; they do not, by themselves, attribute the errors observed in our experiment to a particular cause.

Magnitude–direction decomposition of response vectors. For one state, action pair, and sampling draw, let $e = \Delta y$ be the environment's RGB action response and $w = \Delta \hat{y}$ the model's response, with

D scalar tensor elements. When $\|e\|_2, \|w\|_2 > 0$, define the norm ratio and directional cosine as

$$\rho = \frac{\|w\|_2}{\|e\|_2}, \qquad c = \frac{\langle w, e \rangle}{\|w\|_2 \|e\|_2}. \tag{D.2}$$

By the law of cosines, per-sample response MSE has two equivalent decompositions:

$$\begin{aligned} M_{\text{resp}} &= \frac{\|w - e\|_2^2}{D} \\ &= \frac{\|e\|_2^2}{D}\left(1 + \rho^2 - 2\rho c\right) \\ &= \frac{\|e\|_2^2}{D}\left[(\rho - 1)^2 + 2\rho(1 - c)\right]. \end{aligned} \tag{D.3}$$

The first term in the final expression is the squared norm difference; the second is error due to directional misalignment, scaled by predicted response magnitude ρ. When the model barely responds ($\rho \approx 0$), even a poorly aligned direction contributes little through the directional term. A repair that mainly amplifies the model response also amplifies any remaining directional error. Consequently, a smaller norm-ratio error $|\rho - 1|$ and a larger cosine c do not suffice to guarantee lower M_{resp}. If $w = 0$, c is undefined but $M_{\text{resp}} = \|e\|_2^2/D$; if $e = 0$, both ρ and c are undefined and $M_{\text{resp}} = \|w\|_2^2/D$ should be used directly. With very small $\|w\|_2$, the cosine denominator is also small, so prediction noise or minor pixel changes can substantially alter c. Mean directional cosine at low response magnitudes should therefore be interpreted alongside response norms.

For fixed e and c, scaling the model response along its current direction minimizes Eq. (D.3) over $\rho \geq 0$ at $\rho^* = \max(c, 0)$, rather than generally at 1. In particular, the mismatch difference between a unit-norm-ratio prediction and a zero-response prediction is

$$M_{\text{resp}}(\rho = 1) - M_{\text{resp}}(\rho = 0) = \frac{\|e\|_2^2}{D}(1 - 2c). \tag{D.4}$$

If the directional cosine is below $1/2$, merely restoring the response norm to the environment's level produces larger pixel-level mismatch than predicting no response. This is not a metric error: mismatch requires agreement in *signed pixel changes*, not only equal change magnitude.

Now allow direction and magnitude to change jointly. Write the pre- and post-repair pairs as (ρ_0, c_0) and (ρ_1, c_1), with $\Delta\rho = \rho_1 - \rho_0$ and $\Delta c = c_1 - c_0$. For the same environment response,

$$\begin{aligned} M_1 - M_0 &= \frac{\|e\|_2^2}{D}\left[(\rho_1^2 - \rho_0^2) - 2(\rho_1 c_1 - \rho_0 c_0)\right] \\ &= \frac{\|e\|_2^2}{D}\left[\Delta\rho(\rho_1 + \rho_0 - 2c_1) - 2\rho_0 \Delta c\right]. \end{aligned} \tag{D.5}$$

Thus, the benefit of improved direction must outweigh the error introduced by amplifying a response that remains misaligned. For example, with the same unit-norm environment response, $(\rho_0, c_0) = (0.1, 0)$ and $(\rho_1, c_1) = (0.5, 0.2)$ reduce magnitude error from 0.9 to 0.5 and increase directional cosine from 0 to 0.2, yet normalized mismatch rises from 1.01 to 1.05. This is a mathematical counterexample, not a per-sample measurement from our experiment. The ideal uniform under-response in Section 3.4 has $w = \alpha e$ with $0 < \alpha < 1$, giving $c = 1$ and $M_{\text{resp}} = \|e\|_2^2(1 - \alpha)^2/D$, which necessarily decreases as α approaches 1. Direction-dependent mismatch has no such guarantee.

Cancellation of errors between prediction branches. Equation (D.3) can also be derived from nominal and attacked prediction errors. Let $p_{\text{adv}} = \hat{y}^{\text{adv}} - y^{\text{adv}}$ and $p_{\text{nom}} = \hat{y}^{\text{nom}} - y^{\text{nom}}$. Then $w - e = p_{\text{adv}} - p_{\text{nom}}$, and

$$M_{\text{resp}} = \frac{\|p_{\text{adv}}\|_2^2 + \|p_{\text{nom}}\|_2^2 - 2\langle p_{\text{adv}}, p_{\text{nom}} \rangle}{D}. \tag{D.6}$$

Nominal and attacked prediction MSE constrain only the first two terms. Even if both decrease, response mismatch can rise if previously shared prediction errors between the branches cease to cancel. For example, take the same unit error vector in both branches before repair, so their difference is zero. After repair, reduce each error vector's length to $1/2$ but make them orthogonal: both squared branch-error norms fall from 1 to $1/4$, while the squared response-error norm rises from 0 to $1/2$. This is an equivalent view of the magnitude–direction decomposition, not an additional independent error source.

Scope of interpretation for the aggregate results. In Table 3, mean directional cosine on unseen episodes rises from 0.0459 to 0.1797, mean norm-ratio error falls from 0.9308 to 0.5657, and mean MSE of both prediction branches decreases, whereas mean response RMSE rises from 0.09833 to 0.10222. The identities above show that these trends can coexist, but column means cannot be substituted into per-sample formulas: environment response energy $\|e\|_2^2$ varies by state; ρ, c, and the branch-error inner product are correlated; mean $|\rho - 1|$ does not distinguish under-response from over-response; and mean RMSE does not equal the square root of mean MSE. The aggregates also cannot exclude limited cross-episode generalization or sampling variation as contributors to the higher mismatch. The absolute-difference heatmaps in Figures 5 and D.3 show response magnitude and spatial distribution for selected frames but discard response signs; directional improvement must be assessed with the cosine metric. Response RMSE remains a valid overall measure of pixel-level fidelity, but it cannot alone determine whether response magnitude and direction have separately recovered. To explain mismatch changes on unseen episodes, the terms in Eq. (D.5) or Eq. (D.6) should be reported on the same paired states and sampling draws, alongside task-space or control outcomes.

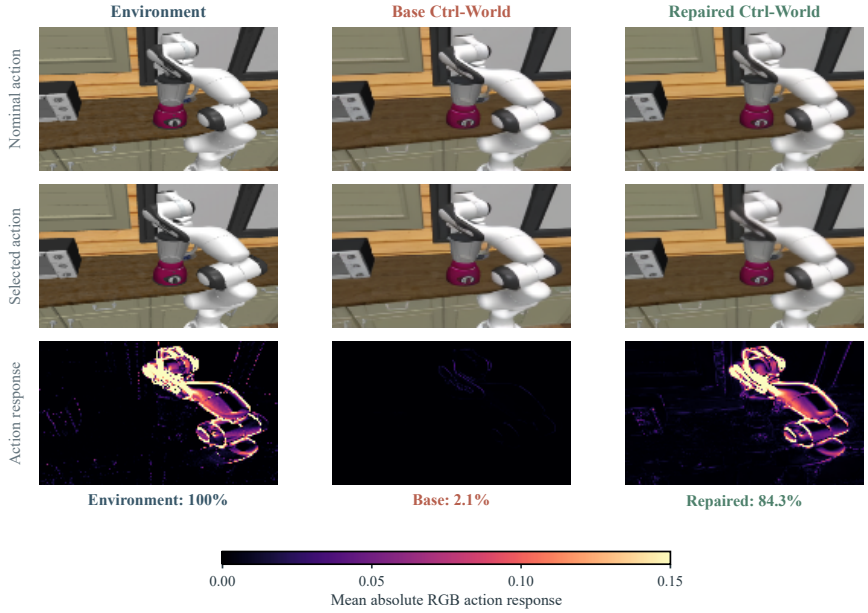


Figure D.3: An action-response repair case from training episode 316. In the displayed view and frame, the norm of the model response relative to the environment rises from 2.1% for Base to 84.3% after repair. This case shows that local response magnitude can recover substantially, but comes from a training episode and does not establish cross-episode repair efficacy.