WORLD MODELS DREAM OF SUCCESS: DIAGNOSING AND REPAIRING FAILURE INSENSITIVITY IN ROBOT WORLD MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Robot world models support policy evaluation, planning, and synthetic data generation, but these applications require predictions that distinguish successful actions from failures. Across four released checkpoints from two architecture families, we observe weak sensitivity to action changes and success-like predictions on verified failures. Although recent work incorporates failures into model training, which data can repair released checkpoints without changing their architecture or training objective still remains underexplored. To this end, we introduce **CureWM**, which constructs alternative actions from successful demonstrations across a severity grid, verifies their outcomes through execution in simulation or on hardware, and fine-tunes released models on the resulting failures and surviving successes alongside nominal demonstrations. This construction provides controlled action contrasts from shared starting contexts. On 484 held-out LIBERO failure counterfactuals, optimism falls from 80% after fine-tuning on the official data to 30–43% across four independently fine-tuned CureWM models (38% mean). In two separate evaluations on a physical robot arm, failure predictions scored as success-like by a latent-distance diagnostic decrease from 90% after fine-tuning on successful demonstrations alone to 33% with CureWM. With failure counts per task, successful replay data, and training budget matched, counterfactual failures yield a success–failure value gap of 0.124, compared with 0.014 for freshly collected on-policy failures. These findings support execution-verified counterfactual replay for post-hoc repair and show why reduced optimism must be evaluated alongside success–failure discrimination.

1 INTRODUCTION

Robot world models learn to predict how environments evolve in response to actions, from latent dynamics models to generative video simulators (Hafner et al., 2025; Yang et al., 2024; NVIDIA, 2025a). In the robotic domain, their predictions support policy evaluation in imagination (Guo et al., 2026), action selection and planning (Kim et al., 2026), and synthetic experience generation for policy learning (Yang et al., 2024; Guo et al., 2026). These applications require more than visually plausible futures: a model must distinguish actions that accomplish the task from those that fail. If failing actions receive high success values or generate success-like futures, decisions based on these predictions can favor actions that do not achieve the intended outcome.

However, how to repair failure prediction in released robot world models still remains underexplored. MiraBench (Yang et al., 2026) found optimism bias in action-conditioned predictions, but diagnosing such failure insensitivity does not tell us which training data could correct it. In particular, failed trajectories need not teach a model how alternative actions from the same context change the outcome. This distinction matters for released checkpoints: simply adding failures may lower predicted success across both successful and failing actions, reducing the optimism without improving discrimination. Therefore, the question is *which kind of data enable post-hoc repair of action-conditioned predictions, and how to distinguish that repair from a general shift toward pessimism.* Concurrent work, such as FACT (Peng et al., 2026), combines failed policy rollouts with action conditioning and failure-aware masking in a world-action model. However, it does not

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.

directly address how to repair released checkpoints while retaining their architecture and training objective, or which failure data are most effective under these constraints.

To fill these knowledge gaps, we introduce **CureWM**, motivated by the observation that correctly predicting the outcome of an observed action does not imply knowing how a different action in the same context would change the outcome. CureWM perturbs successful demonstrations across a severity grid while preserving actions before intervention, verifies outcomes through execution, and retains failed and successful replays for post-training. The architecture and training objective remain unchanged. We evaluate four released checkpoints from two architecture families, which are Cosmos Policy (Kim et al., 2026) and Ctrl-World (Guo et al., 2026), and test CureWM in simulation and on a physical robot arm. The ALOHA checkpoint is evaluated for diagnosis only (Appendix C). Experiments on LIBERO (Liu et al., 2023) and a physical robot show improved success–failure discrimination. Matched comparisons favor counterfactual failures over fresh policy failures, while ablations examine successful replays and verified outcome labels. These results also show why reduced optimism alone does not establish repair. Our contributions are as follows:

1. We introduce CureWM, which repairs released world models using execution-verified counterfactual replays without changing their architecture or training objective.
2. We demonstrate repair in simulation and on a physical robot, including improved success–failure discrimination and failure detection. Matched LIBERO comparisons show stronger discrimination than training on fresh policy failures.
3. We show that reduced optimism alone can overstate repair, and evaluate success–failure discrimination together with false alarms on successful actions.

2 Related Work

World models and action-conditioned reliability. Cosmos Policy (Kim et al., 2026) predicts actions, future observations, and values for control and planning, while Ctrl-World (Guo et al., 2026) supports policy evaluation and synthetic trajectory generation through action-conditioned video prediction. MiraBench (Yang et al., 2026) evaluates physics adherence, action following, and optimism bias, and provides the perturbation families used in our experiments. WorldArena (Shang et al., 2026) likewise distinguishes perceptual quality from functional utility. We study post-hoc repair. Our support argument relates to causal confusion in imitation learning (de Haan et al., 2019), but does not identify a causal mechanism inside the model.

Failure and counterfactual supervision. Learning from unsuccessful behavior and constructed examples has been widely studied: Dex-Net (Mahler et al., 2017) learns grasp robustness from synthetic examples with analytic labels, HER (Andrychowicz et al., 2017) reuses unsuccessful attempts through goal relabeling, and CoDA (Pitis et al., 2020) constructs counterfactual transitions using locally factored dynamics. AHA (Duan et al., 2025) synthesizes manipulation failures to train a vision-language model for failure detection and natural-language reasoning. Concurrent FACT (Peng et al., 2026) incorporates failed rollouts into world-action model training. We do not claim novelty in adding failure examples. CureWM combines controlled perturbations, execution-verified outcomes, and successful replay partners to repair released checkpoints with their original architecture and objective. Furthermore, matched comparisons test which data choices improve repair.

Conservative values, failure detection, and predictive repair. Offline RL addresses unreliable extrapolation through data-constrained action selection (Fujimoto et al., 2019), conservative value learning (Kumar et al., 2020), and uncertainty-aware penalties or pessimistic dynamics (Yu et al., 2020; Kidambi et al., 2020). These methods motivate caution about unsupported actions, but conservative decision-making and accurate outcome discrimination are distinct objectives. Our label ablation illustrates that reduced optimism can accompany underestimated success predictions. It does not evaluate those offline RL algorithms. Foresight (Zhang et al., 2026a) instead learns failure detection from action-conditioned world-model representations. CureWM updates the world model's own predictions. We assess whether it better distinguishes successful from failing actions, rather than simply assigning lower success values to both.

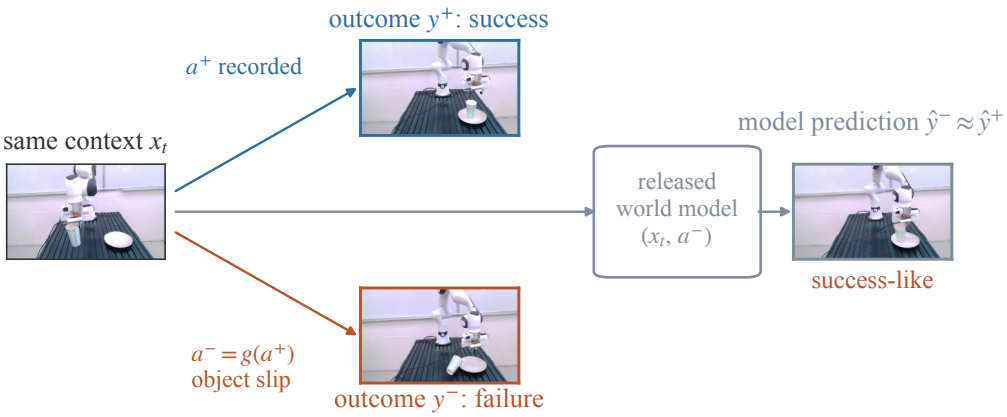


Figure 1: **Failure insensitivity on a held-out Franka counterfactual.** From the same starting context, the original action sequence succeeds and a perturbed sequence fails under physical execution, yet the model predicts a success-like future for the failing sequence.

3 METHOD

3.1 FAILURE INSENSITIVITY

Let x_t denote the observation history and task context, and let $a^+_{t:T}$ and $a^-_{t:T}$ denote action sequences that succeed and fail from the same starting context. We call a model *failure-insensitive* when its predictions fail to distinguish these outcomes. Figure 1 illustrates this behavior. This differs from optimism alone: a model can assign similar values to successful and failing actions even when neither value is high. We therefore evaluate both predictions on failing actions and discrimination between successful and failing actions.

3.2 MEASURING FAILURE SENSITIVITY

Value prediction. For a model predicting native value predictions $V_\theta(x, a)$, we measure the mean *success–failure value gap* over N action pairs:

$$\Delta_{\mathrm{SF}} = \frac{1}{N} \sum_{i=1}^{N} \left[V_\theta(x_i, a_i^+) - V_\theta(x_i, a_i^-) \right]. \tag{1}$$

A positive gap means successful actions receive higher values on average. We report both values to check whether improved discrimination comes at the cost of underestimating successful actions. We use 0.5 as a fixed diagnostic threshold for the native value prediction. We do not assume that this value is a calibrated success probability. False-positive rate measures successful replays labeled as failures by this threshold. We define *value optimism* as the fraction of verified failing sequences with $V_\theta(x_i, a_i^-) > \tau$, using $\tau = 0.5$ on LIBERO. Including successful perturbed replays, we also report AUROC (Fawcett, 2006) and the fraction of successful replays incorrectly predicted to fail.

Video prediction. Let $\hat{z}_T(a)$ be the predicted image representation, and let y_T^+ and y_T^- be recorded successful and failed outcomes for the corresponding action pair. With the frozen encoder ϕ, we define

$$s(a) = d\big(\hat{z}_T(a), \phi(y_T^-)\big) - d\big(\hat{z}_T(a), \phi(y_T^+)\big), \tag{2}$$

where d is mean squared distance between image representations, averaged across the camera views used by the model. Positive scores indicate predictions closer to the successful reference. We report *visual optimism* which is the fraction of failing sequences with $s(a^-) > 0$, the mean failure-action score, the mean paired difference $\Delta_{\mathrm{SF}} = \frac{1}{N}\sum_i [s(a_i^+) - s(a_i^-)]$, and pair-ranking accuracy, the fraction with $s(a_i^+) > s(a_i^-)$. These are offline diagnostics requiring recorded reference outcomes.

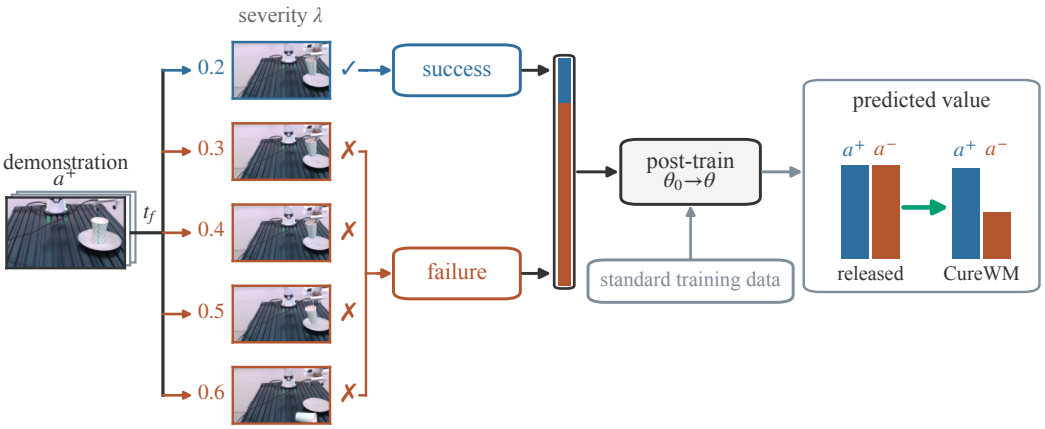


Figure 2: **The overview of CureWM.** We perturb successful demonstrations at different severities, verify outcomes through execution, and use both successful and failing replays for post-training alongside the original training data. Severity values, outcome assignments, and predicted-value bars are used for illustration rather than measured results.

3.3 CUREWM

CureWM constructs execution-verified action alternatives for post-training (Figure 2). It is motivated by the observation that, when only one action is observed per context, fitting its outcome does not require distinguishing the consequences of alternative actions. Proposition 1 in Appendix B formalizes this observation.

Controlled counterfactual construction. We represent a successful demonstration as $\xi^+ = (x_0, o_{0:T}, a^+_{0:T-1}, \ell = 1)$, where x_0 is the initial context and ℓ indicates task success. We identify approach, grasp, carry, and place phases and select an intervention time t_f for perturbation family f. Following MiraBench (Yang et al., 2026), we use approach overshoot, carry slip, contact oscillation, insufficient grip, premature release, and wrist tilt where applicable. For severity $\lambda \in \Lambda_f$, we construct

$$a^{f,\lambda}_{0:T-1} = g_{f,\lambda}\big(a^+_{0:T-1}\big)\,, \qquad a^{f,\lambda}_t = a^+_t \quad \text{for } t < t_f. \tag{3}$$

The severity grid Λ_f produces multiple action alternatives while preserving the actions before intervention. We do not label these alternatives as failures before execution.

Execution-based outcome verification. We execute each modified sequence from the corresponding initial context in simulation or on hardware, recording its observations, executed actions, and task outcome $\ell \in \{0, 1\}$. A perturbation family enters training only when its empirical failure rate satisfies $\hat{p}_f(\lambda_{j+1}) \geq \hat{p}_f(\lambda_j)$ for adjacent increasing severities. This condition applies to aggregate failure rates, not to every demonstration individually. We retain successful and failing replays as $\mathcal{D}^+_{\text{rep}}$ and $\mathcal{D}^-_{\text{rep}}$, respectively. Individual families need not produce both outcomes at the tested severities.

Post-training with verified trajectories. We convert replays to the released model's training format using $\mathcal{C}$ and combine them with its base training data:

$$\mathcal{D}_{\text{cure}} = \mathcal{D}_{\text{base}} \uplus \mathcal{C}(\mathcal{D}^+_{\text{rep}}) \uplus \mathcal{C}(\mathcal{D}^-_{\text{rep}}). \tag{4}$$

Here $\mathcal{D}_{\text{base}}$ contains the demonstrations and official rollouts used for the corresponding model, and $\uplus$ combines the data without removing duplicates. For models with a value head, failed replays receive zero value targets. Successful replays retain the original labeling rule, which discounts the final success reward according to the remaining steps. For video prediction, supervision comes from the observations recorded during execution.

Starting from released parameters θ_0, we fine-tune the full prediction network using its original objective:

$$\min_\theta \; \mathbb{E}_{u \sim q_{\text{cure}}} \left[\mathcal{L}_{\text{native}}(\theta; u)\right], \qquad \theta \leftarrow \theta_0 \quad \text{at initialization}, \tag{5}$$

Table 1: **LIBERO prediction and robot task success.** Baseline and CureWM use the same fine-tuning steps. Suite counts indicate failed/successful held-out replays. False-positive rate measures successful replays assigned a value at or below 0.5. Means weight the four suites equally.

Suite	Model	Value optimism ↓	Δ_{SF} ↑	AUROC ↑	False-positive rate ↓	Task success ↑
Goal (484/1196)	Released	79.13	$+0.002$	0.497	30.5	98.4
	Baseline	79.55	$+0.003$	0.495	31.0	96.4
	CureWM	30.17	$+0.282$	0.743	38.0	94.8
Spatial (367/443)	Released	65.67	-0.014	0.518	36.6	98.4
	Baseline	64.03	-0.011	0.522	36.3	95.6
	CureWM	47.68	$+0.081$	0.629	35.0	96.2
Object (286/524)	Released	54.55	$+0.004$	0.464	57.4	99.8
	Baseline	53.85	$+0.004$	0.466	57.4	99.0
	CureWM	37.41	$+0.086$	0.587	59.4	99.8
LIBERO-10 (359/331)	Released	2.51	-0.001	0.479	99.7	98.0
	Baseline	2.51	-0.001	0.483	99.1	96.2
	CureWM	2.51	$+0.003$	0.493	99.4	95.4
Mean	Released	50.47	-0.002	0.490	56.1	98.65
	Baseline	49.99	-0.001	0.492	56.0	96.80
	CureWM	29.44	$+0.113$	0.613	58.0	96.55

where u is a training example and q_{cure} follows the model's existing sampling procedure on $\mathcal{D}_{\mathrm{cure}}$. Successful and failing trajectories are related through their construction, but are not supplied as explicit training pairs. We introduce neither a new prediction head nor a pairwise loss. Training and sampling settings are provided in Appendix D.2.

4 Simulation Experiments

4.1 LIBERO

4.1.1 Evaluation Setup

The baseline and CureWM use official data from all four LIBERO suites; CureWM additionally uses counterfactual replays from LIBERO-Goal. Goal evaluation contains 484 verified failures and 1,196 successful perturbed replays from two independently generated sets using the training perturbation families and severity grid. Their source demonstrations are excluded from counterfactual construction but mostly present in the official data. The second set was generated after the compared models finished training. Spatial, Object, and LIBERO-10 test transfer without counterfactual training replays from those suites. Robot task success uses 500 episodes per suite and model.

4.1.2 Results

We compare CureWM with the released model and a baseline fine-tuned on official data for the same number of steps. The detailed numerical results are presented in Table 1. On LIBERO-Goal, CureWM reduces value optimism from the baseline's 79.55% to 30.17%, increases Δ_{SF} from 0.003 to 0.282, and improves AUROC from 0.495 to 0.743. However, the proportion of successful replays incorrectly predicted to fail increases from 31.0% to 38.0%. CureWM improves discrimination, but the fixed-threshold detector predicts more successful replays as failures. Without counterfactual training replays from LIBERO-Spatial or LIBERO-Object, CureWM improves their AUROC from 0.522 and 0.466 to 0.629 and 0.587, respectively, alongside larger value gaps and lower optimism. On LIBERO-10, AUROC remains near chance at 0.493. At the fixed threshold of 0.5, low optimism of 2.51% coexists with a 99.4% false-positive rate. Thus, low optimism does not indicate reliable failure detection on this suite. Mean task success is 98.65% for Released, 96.80% for the baseline, and 96.55% for CureWM, whose changes relative to the baseline vary by suite (Table 1).

Table 2: **RoboCasa prediction sensitivity and task success.** All Δ_{SF} values use the same 120 modified-action examples, whose modified actions are not re-executed. AUROC is held-out articulated failure detection on 225 rollouts (111 successes, 114 failures) scored at 75% of episode length. Task success is out of 30 episodes (three evaluation seeds, ten episodes each). Expanded replays add drawer and door tasks. Masking removes action-prediction loss on failed replays.

Training data / setting	Steps	AUROC ↑	Δ_{SF} ↑	OpenDrawer	CloseDoubleDoor
Released	—	0.634	0.0170	28/30	30/30
Grasp replays	5k	0.674	0.0672	17/30	23/30
Grasp replays	20k	0.677	0.0342	26/30	29/30
Expanded replays	5k	0.839	0.0302	25/30	26/30
Expanded replays	20k	0.780	0.0193	25/30	28/30
Grasp replays, action loss masked	5k	0.662	0.0308	20/30	30/30

Additional experiments assess training variability and discrimination within perturbation types. Across four independently fine-tuned CureWM models, Goal value optimism ranges from 30.2% to 43.0%, averaging 37.9% (Appendix D.2). Within 21 eligible perturbation-family–severity groups, mean AUROC increases from 0.537 to 0.714. The improvement therefore cannot be explained by perturbation type or severity alone (Appendix D.3). Experiments excluding individual perturbation types from counterfactual training show uneven transfer (Appendix D.4).

4.2 RoboCasa

4.2.1 Evaluation Setup

We evaluate failure detection on 111 successful and 114 failing drawer and door manipulation rollouts excluded from training, reporting AUROC from predicted task-success values at 75% of episode length. Separately, we measure Δ_{SF} on 120 examples with modified actions from official rollouts. These actions are not re-executed, so this evaluation measures prediction sensitivity rather than detection of verified failures. Robot task success is evaluated on OpenDrawer and CloseDoubleDoor from RoboCasa (Nasiriany et al., 2024) over 30 episodes per task and model, using three evaluation seeds with ten episodes each.

4.2.2 Results

Expanded training achieves the highest held-out detection AUROC at 5,000 steps: 0.839 versus 0.634 for the released model (Table 2). Grasp-focused training instead produces the largest value gap, 0.0672, but reduces OpenDrawer and CloseDoubleDoor success from 28/30 and 30/30 to 17/30 and 23/30. Prediction sensitivity, failure detection, and task success therefore do not improve together.

Adding drawer and door replays improves detection and task success relative to grasp-focused training at 5,000 steps, while changing both coverage and data quantity. Its task-success advantage does not persist at 20,000 steps. Masking failed-action prediction loss also improves task success at 5,000 steps, but reduces the value gap and does not improve detection AUROC relative to unmasked training. Further comparisons are provided in Appendix E.

4.3 Ctrl-World

4.3.1 Evaluation Setup

Starting from the same LIBERO-adapted Ctrl-World checkpoint, we compare CureWM with fine-tuning on unperturbed successful demonstrations. We evaluate predictions over four interaction steps (16 raw frames) using Equation 2 and frozen SVD (Blattmann et al., 2023) VAE representations from three camera views. The evaluation contains 30 action pairs, including ten excluded from counterfactual training. The full set partially overlaps training.

Table 3: Ctrl-World: full-pool and held-out evaluation.

	Full pool: 30 pairs		Held-out subset: 10 pairs	
Model	Optimism	Mean $s(a^-)$	Optimism	Mean $s(a^-)$
Adapted	27/30	+0.080	9/10	+0.076
Control	26/30	+0.069	8/10	+0.071
CureWM	21/30	+0.016	6/10	+0.005

Table 4: Cup pick-and-place evaluation on Franka Research 3. AUROC uses both successful and failed reference outcomes.

Model	Optimism	Pair-ranking accuracy (%)	Δ_{SF}	AUROC
First hardware evaluation				
Adapted	13/15	80.0	+0.077	0.75
Baseline	15/15	66.7	+0.066	0.65
CureWM, model 1	6/15	100.0	+0.251	0.95
CureWM, model 2	6/15	100.0	+0.271	1.00
Second hardware evaluation				
Adapted	15/15	93.3	+0.05	0.75
Baseline	12/15	66.7	+0.06	0.70
CureWM, model 1	4/15	100.0	+0.19	0.97
CureWM, model 2	6/15	93.3	+0.17	0.96

4.3.2 Results

On the ten held-out pairs, CureWM reduces visual optimism from 8/10 to 6/10 and the mean failure-action score from 0.071 to 0.005 (Table 3). The paired analysis supports the score reduction, but the optimism-rate difference is not statistically established. Full-set results provide supporting evidence because of their training overlap. A second independently fine-tuned model improves the full-set score in the same direction, without a statistically established difference ($p \approx .18$). Protocols and paired analyses are provided in Appendix F.

5 Repair on a Physical Robot

5.1 Evaluation Setup

We evaluate CureWM on a Franka Research 3 using DROID-pretrained Ctrl-World (Khazatsky et al., 2024; Guo et al., 2026). Cup pick-and-place is the primary repair task. Cube picking tests task-specific repair and transfer from cup-task repair. Replay outcomes are verified through physical execution. After 1,400 steps of scene adaptation using successful demonstrations, we fine-tune CureWM on 90 failed replays, 24 successful wrist-tilt replays, and 31 unperturbed successful trajectories. The baseline uses only the same 31 unperturbed trajectories. Both receive 3,600 additional training steps, matching training duration but not distinct-data counts. Successful perturbed replays come only from wrist tilt and the other three tested perturbation types produce no successful replays at the evaluated severities. Data collection and splits are detailed in Appendix G.1. Each of two hardware evaluations contains 15 failing action sequences paired with their original successful sequences from ten held-out demonstrations. These demonstrations and their perturbed replays are excluded from training. We measure visual optimism and success–failure discrimination using Equation 2. We also measure failure-detection AUROC using distance to the image recorded at the end of each successful execution. Both diagnostics require recorded reference outcomes.

5.2 Results

After success-only scene adaptation, 65 of 71 verified cup-task failures receive success-like predictions under the latent-distance diagnostic. This is not a human-judged success rate, and most of these examples subsequently enter repair training. Repair is evaluated separately on held-out demonstra-

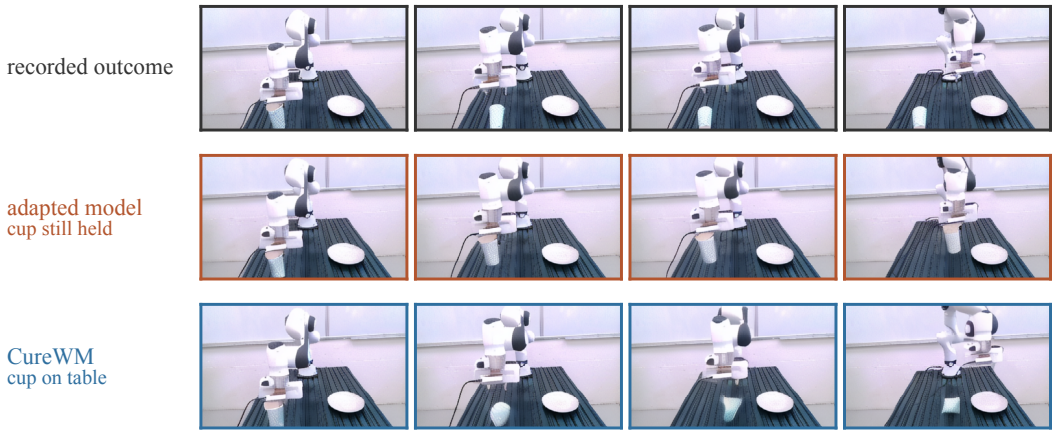


Figure 3: **Predicted futures for a failing cup-task action sequence.** Columns show successive time steps. Rows show recorded execution, the scene-adapted model, and CureWM. CureWM predicts the observed release; the adapted model predicts continued grasping.

tions. In the first evaluation, CureWM reduces visual optimism from the baseline's 15/15 to 6/15 (Table 4). The mean failure-action score changes from $+0.184$ to -0.027, while the success-action score changes from $+0.249$ to $+0.224$. The larger change therefore occurs on failing actions. Before the second evaluation, all models receive the same additional adaptation on successful demonstrations and pass the rendering-quality check (Appendix G.2). Across both evaluations, visual optimism decreases from 27/30 to 10/30, and failure-detection AUROC using successful outcomes as references increases from the baseline's 0.67 to 0.90. Adding only the 24 successful wrist-tilt replays to the baseline leaves first-evaluation optimism at 15/15, although its mean failure-action score decreases from 0.184 to 0.138. These additional successful replays do not reproduce CureWM's improvement. Figure 3 illustrates the change in generated futures for a failing cup-task action sequence.

Three non-author raters assessed blinded predictions from the first evaluation. For the 15 failing actions, majority judgments classify 8/15 adapted-model predictions and 0/15 CureWM predictions as successful, compared with 13/15 and 6/15 under the distance-based diagnostic. Human judgments support the direction of improvement while showing that the diagnostic more frequently classifies predictions as successful (Appendix G.3). On cube picking, task-specific repair reduces the mean failure-action score relative to the baseline by 0.025 on 29 held-out pairs from ten scenes. Applying cup-task repair before cube-scene adaptation instead gives a difference of -0.007. The results support task-specific repair but do not establish transfer from cup to cube picking (Appendix G.4).

6 Discussion

6.1 Counterfactual Failures versus Policy Failures

On LIBERO, we compare 140 fresh policy failures with 140 counterfactual failures, matching per-task failure counts, the same 580 successful replays, and 7,500 training steps. On 484 held-out counterfactuals, their value gaps are 0.014 and 0.124, respectively (Table 5). The comparison is separate from the 17,500-step experiments. Of the selected policy failures, 69 have successful rollouts from the same task and initial-state index, but their actions differ from the first step. Counterfactual replay preserves actions before intervention. Failure type, timing, and coverage also differ, so shared prefixes are not established as the sole explanation. Up-sampling official failures leaves optimism at 78.3%, but does not match coverage, novelty, or repetition. Neither comparison establishes that policy failures are ineffective in general (Appendix D.5).

6.2 Successful Replays from the Same Demonstrations

The *no-graded* variant removes the 580 successful perturbed replays, while *other-demo* replaces them with successful replays from different demonstrations, approximately matched by perturbation family and severity. Both retain the 242 verified failures and original demonstration data. Across

Table 5: **Failure sources and successful replays on LIBERO.** All models are evaluated on the same 484 held-out failing replays and their successful counterparts. The 7,500-step comparison matches failure counts and successful training replays. The 17,500-step comparisons report the primary models. The two groups use different failure pools and training budgets. Δ_{SF} is computed before rounding.

Training setting	Value optimism (%) ↓	Mean $V(a^-)$	Mean $V(a^+)$	Δ_{SF} ↑
Released model	79.1	0.601	0.603	+0.002
Matched failure-source comparison: 7,500 steps				
Official-data baseline	79.3	0.594	0.597	+0.003
Fresh policy failures	66.9	0.545	0.559	+0.014
Matched counterfactual failures	46.9	0.463	0.587	+0.124
Training-data comparisons: 17,500 steps				
Official-data baseline	79.5	0.602	0.605	+0.003
Own-failure up-sampling	78.3	0.599	0.603	+0.004
No-graded	49.0	0.441	0.593	+0.152
Other-demo	60.1	0.495	0.603	+0.107
CureWM	30.2	0.300	0.582	+0.282

two independently fine-tuned models per variant, value optimism is 43.4–49.0% for no-graded and 44.8–60.1% for other-demo, compared with 30.2–43.0% across four full CureWM models. These results support an additional contribution from successful replays of the same demonstrations. However, the smallest observed difference is only 0.4 percentage points, and differences in task composition, sampling frequency, and training variation limit attribution to demonstration identity alone (Appendix D.5).

6.3 Lower Optimism Does Not Always Mean Better Predictions

Fresh policy failures reduce optimism by 12.4 percentage points while lowering values for both successful and failing actions (Table 5). Labeling all perturbed replays as failures, including 580 successful replays, also lowers training optimism but reduces the success-side value to 0.147. Its held-out value gap is 0.103 versus CureWM's 0.282 (Appendix D.5). These comparisons motivate evaluating discrimination and fixed-threshold false-positive rates alongside optimism.

6.4 Limitations

Only 53 of 484 LIBERO-Goal failures come from demonstrations absent from the shared official data. Transfer across perturbation types and objects is inconsistent, and the prediction horizon limits observable failure consequences. Cosmos Policy's video repair remains partial. Hardware evidence covers two objects on one robot, with small held-out samples and unequal training-data counts.

CureWM requires executable counterfactual actions and verifiable outcomes; hardware detection also requires episode-specific references. Predictive gains do not establish improved control: RoboCasa exhibits a repair–control trade-off, best-of-N selection shows no statistically established benefit, and policy ranking is untested.

7 Conclusion

We introduce CureWM, which repairs released robot world models using execution-verified counterfactual replays without changing their architecture or training objective. Experiments in simulation and on hardware show improved success–failure discrimination. Matched comparisons favor counterfactual failures over fresh policy failures at the tested budget, while ablations support retaining successful replays from the same demonstrations. These results also show why lower optimism alone does not establish repair. Broader transfer and improved robot control remain open challenges.

AI USE STATEMENT

We used generative AI (i.e., Claude and ChatGPT) tools to assist with drafting, revising, and polishing the manuscript, and to assist in proving the mathematical claims presented in Appendix B. The authors reviewed the AI-assisted text for accuracy and consistency with the experimental evidence. We also checked the mathematical arguments, including their assumptions, logical steps, and the scope of their conclusions, and revised them where necessary. We take responsibility for the final content of this work, including all AI-assisted text, mathematical claims, and proofs.

REPRODUCIBILITY STATEMENT

We provide anonymized code and analysis scripts in the supplementary materials. Section 3 describes counterfactual construction, execution-based outcome verification, and post-training. Appendix D.2 documents training settings, sampling configuration, and evaluated checkpoints. Data composition and evaluation splits are detailed in Appendices D.1, E, F, and G.1. Hardware evaluation, human assessment, and cube-task protocols are provided in Appendices G.2, G.3, and G.4, respectively. The assumptions and proofs underlying our theoretical argument appear in Appendix B.

REFERENCES

Marcin Andrychowicz, Filip Wolski, Alex Ray, Jonas Schneider, Rachel Fong, Peter Welinder, Bob McGrew, Josh Tobin, Pieter Abbeel, and Wojciech Zaremba. Hindsight experience replay. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

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

Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, Varun Jampani, and Robin Rombach. Stable Video Diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023.

Qingwen Bu, Jisong Cai, Li Chen, Xiuqi Cui, Yan Ding, Siyuan Feng, Xindong He, Xu Huang, et al. AgiBot World Colosseo: A large-scale manipulation platform for scalable and intelligent embodied systems. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2025. arXiv:2503.06669.

Josef Chen. AEGIS: A backup reflex for physical AI. *arXiv preprint arXiv:2606.06660*, 2026.

Pim de Haan, Dinesh Jayaraman, and Sergey Levine. Causal confusion in imitation learning. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Jiafei Duan, Wilbert Pumacay, Nishanth Kumar, Yi Ru Wang, Shulin Tian, Wentao Yuan, Ranjay Krishna, Dieter Fox, Ajay Mandlekar, and Yijie Guo. AHA: A vision-language-model for detecting and reasoning over failures in robotic manipulation. In *International Conference on Learning Representations*, 2025.

Tom Fawcett. An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8):861–874, 2006. doi: 10.1016/j.patrec.2005.10.010.

Scott Fujimoto, David Meger, and Doina Precup. Off-policy deep reinforcement learning without exploration. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2052–2062. PMLR, 2019.

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

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

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse control tasks through world models. *Nature*, 640(8059):647–653, 2025. doi: 10.1038/s41586-025-08744-2.

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

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.

Alexander Khazatsky, Karl Pertsch, Suraj Nair, Ashwin Balakrishna, Sudeep Dasari, Siddharth Karamcheti, Soroush Nasiriany, Mohan Kumar Srirama, Lawrence Yunliang Chen, Kirsty Ellis, Peter David Fagan, Joey Hejna, Masha Itkina, Marion Lepert, Yecheng Jason Ma, Patrick Tree Miller, Jimmy Wu, Suneel Belkhale, Shivin Dass, Huy Ha, Arhan Jain, Abraham Lee, Youngwoon Lee, Marius Memmel, Sungjae Park, Ilija Radosavovic, Kaiyuan Wang, Albert Zhan, Kevin Black, Cheng Chi, Kyle Beltran Hatch, Shan Lin, Jingpei Lu, Jean Mercat, Abdul Rehman, Pannag R Sanketi, Archit Sharma, Cody Simpson, Quan Vuong, Homer Rich Walke, Blake Wulfe, Ted Xiao, Jonathan Heewon Yang, Arefeh Yavary, Tony Z. Zhao, Christopher Agia, Rohan Baijal, Mateo Guaman Castro, Daphne Chen, Qiuyu Chen, Trinity Chung, Jaimyn Drake, Ethan Paul Foster, Jensen Gao, David Antonio Herrera, Minho Heo, Kyle Hsu, Jiaheng Hu, Donovon Jackson, Charlotte Le, Yunshuang Li, Roy Lin, Zehan Ma, Abhiram Maddukuri, Suvir Mirchandani, Daniel Morton, Tony Nguyen, Abigail O'Neill, Rosario Scalise, Derick Seale, Victor Son, Stephen Tian, Emi Tran, Andrew E. Wang, Yilin Wu, Annie Xie, Jingyun Yang, Patrick Yin, Yunchu Zhang, Osbert Bastani, Glen Berseth, Jeannette Bohg, Ken Goldberg, Abhinav Gupta, Abhishek Gupta, Dinesh Jayaraman, Joseph J Lim, Jitendra Malik, Roberto Martín-Martín, Subramanian Ramamoorthy, Dorsa Sadigh, Shuran Song, Jiajun Wu, Michael C. Yip, Yuke Zhu, Thomas Kollar, Sergey Levine, and Chelsea Finn. DROID: A large-scale in-the-wild robot manipulation dataset. In *Proceedings of Robotics: Science and Systems*, 2024. doi: 10.15607/RSS.2024.XX.120.

Rahul Kidambi, Aravind Rajeswaran, Praneeth Netrapalli, and Thorsten Joachims. MOReL: Model-based offline reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 33, 2020.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan P Foster, Pannag R Sanketi, Quan Vuong, Thomas Kollar, Benjamin Burchfiel, Russ Tedrake, Dorsa Sadigh, Sergey Levine, Percy Liang, and Chelsea Finn. OpenVLA: An open-source vision-language-action model. In *Proceedings of The 8th Conference on Robot Learning*, volume 270 of *Proceedings of Machine Learning Research*, pp. 2679–2713. PMLR, 2025. URL https://proceedings.mlr.press/v270/kim25c.html.

Moo Jin Kim, Yihuai Gao, Tsung-Yi Lin, Yen-Chen Lin, Yunhao Ge, Grace Lam, Percy Liang, Shuran Song, Ming-Yu Liu, Chelsea Finn, and Jinwei Gu. Cosmos Policy: Fine-tuning video models for visuomotor control and planning. In *International Conference on Learning Representations*, 2026.

Aviral Kumar, Aurick Zhou, George Tucker, and Sergey Levine. Conservative Q-learning for offline reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 33, 2020.

Dayou Li, Jiuzhou Lei, Hao Wang, Lulin Liu, Yunhao Yang, Zihan Wang, Bangya Liu, Minghui Zheng, and Zhiwen Fan. Learning actionable manipulation recovery via counterfactual failure synthesis. *arXiv preprint arXiv:2603.13528*, 2026.

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. Datasets and Benchmarks Track.

Jeffrey Mahler, Jacky Liang, Sherdil Niyaz, Michael Laskey, Richard Doan, Xinyu Liu, Juan Aparicio Ojea, and Ken Goldberg. Dex-Net 2.0: Deep learning to plan robust grasps with synthetic point clouds and analytic grasp metrics. In *Proceedings of Robotics: Science and Systems*, 2017.

Soroush Nasiriany, Abhiram Maddukuri, Lance Zhang, Adeet Parikh, Aaron Lo, Abhishek Joshi, Ajay Mandlekar, and Yuke Zhu. RoboCasa: Large-scale simulation of household tasks for generalist robots. In *Proceedings of Robotics: Science and Systems*, 2024. doi: 10.15607/RSS.2024.XX.050.

NVIDIA. Cosmos world foundation model platform for physical AI. *arXiv preprint arXiv:2501.03575*, 2025a.

NVIDIA. GR00T N1: An open foundation model for generalist humanoid robots. *arXiv preprint arXiv:2503.14734*, 2025b.

Quanquan Peng, Yutong Liang, Rui Yan, Nicklas Hansen, and Xiaolong Wang. FACT: Failure-aware causal training for world-action models. *arXiv preprint arXiv:2608.10232*, 2026.

Silviu Pitis, Elliot Creager, and Animesh Garg. Counterfactual data augmentation using locally factored dynamics. In *Advances in Neural Information Processing Systems*, volume 33, 2020.

Bo-Kai Ruan, Teng-Fang Hsiao, Ling Lo, and Hong-Han Shuai. Is the future compatible? Diagnosing dynamic consistency in world action models. *arXiv preprint arXiv:2605.07514*, 2026.

Yu Shang, Zhuohang Li, Yiding Ma, Weikang Su, Xin Jin, Ziyou Wang, Lei Jin, Xin Zhang, Yinzhou Tang, Haisheng Su, Chen Gao, Wei Wu, Xihui Liu, Dhruv Shah, Zhaoxiang Zhang, Zhibo Chen, Jun Zhu, Yonghong Tian, Tat-Seng Chua, Wenwu Zhu, and Yong Li. WorldArena: A unified benchmark for evaluating perception and functional utility of embodied world models. *arXiv preprint arXiv:2602.08971*, 2026.

Sherry Yang, Yilun Du, Kamyar Ghasemipour, Jonathan Tompson, Leslie Kaelbling, Dale Schuurmans, and Pieter Abbeel. Learning interactive real-world simulators. In *International Conference on Learning Representations*, 2024.

Tianzhuo Yang, Zihan Shen, Zirui Mi, Zhaoyi Zhang, Jiayi Zhou, Jiaming Ji, Juntao Dai, Jiawei Chen, Boyuan Chen, and Yaodong Yang. MiraBench: Evaluating action-conditioned reliability in robotic world models. *arXiv preprint arXiv:2605.29360*, 2026.

Tianhe Yu, Garrett Thomas, Lantao Yu, Stefano Ermon, James Zou, Sergey Levine, Chelsea Finn, and Tengyu Ma. MOPO: Model-based offline policy optimization. In *Advances in Neural Information Processing Systems*, volume 33, 2020.

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

Yanyan Zhang, Disheng Liu, Kai Ye, Chaoda Song, Xinpeng Li, Mohsen Hariri, Vikash Singh, Yu Yin, and Vipin Chaudhary. Imagining recovery: Inference-time counterfactual realignment for vision-language-action models. *arXiv preprint arXiv:2608.14822*, 2026b.

Tony Z. Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning fine-grained bimanual manipulation with low-cost hardware. In *Proceedings of Robotics: Science and Systems*, 2023. doi: 10.15607/RSS.2023.XIX.016.

A Extended Related Work

World models and their evaluation. Latent world models support learning through imagined interaction (Ha & Schmidhuber, 2018; Hafner et al., 2025), while video simulators extend prediction to visually rich environments (Yang et al., 2024; NVIDIA, 2025a; Guo et al., 2026). Their potential use with generalist policies (Kim et al., 2025; Black et al., 2025; NVIDIA, 2025b) makes action-conditioned reliability important. MiraBench (Yang et al., 2026) provides the perturbation families used in our experiments. Complementary evaluations examine embodied executability (Jiang et al., 2026) and dynamic consistency (Ruan et al., 2026). Research on predictable world-model hallucinations connects diagnosis to data coverage and mitigation (Hansen & Wang, 2026). These works

motivate evaluating consequences beyond visual plausibility. CureWM studies the response of released checkpoints to execution-verified counterfactual data.

Failure data, detection, and recovery. Large manipulation datasets provide a broader data context (Bu et al., 2025). Failure judgments, detection, backup behavior, and counterfactual recovery are studied by AHA (Duan et al., 2025), Foresight (Zhang et al., 2026a), AEGIS (Chen, 2026), and CoRe (Zhang et al., 2026b). Counterfactual failure synthesis also supports recovery learning (Li et al., 2026). Our intervention changes the world model's predictions through post-training. It is distinct from fitting a separate detector or selecting a recovery action, although these approaches can be complementary.

Relation to FACT and negative-data learning. FACT (Peng et al., 2026) combines failure data with world-action-model design and failure-aware masking. CureWM instead retains released architectures and their original objectives in its main treatment. Our matched fresh-failure experiment compares two data constructions within this fixed recipe and it is not an evaluation of FACT. More generally, learning from unsuccessful attempts, relabeled goals, and counterfactual augmentation has substantial work (Mahler et al., 2017; Andrychowicz et al., 2017; Pitis et al., 2020). Our contribution concerns the construction and evaluation of post-hoc repair data, not the introduction of negative examples. The support argument is also related to concerns about causal confusion (de Haan et al., 2019), without identifying a particular mechanism in the trained network. Conservative offline learning provides a separate context for distinguishing lower values from better discrimination (Fujimoto et al., 2019; Kumar et al., 2020; Yu et al., 2020; Kidambi et al., 2020).

B Theoretical Motivation

B.1 Setting and Scope

Let $D = \{(x_i, a_i, y_i)\}_{i=1}^n$ contain contexts, actions, and binary outcomes $y_i \in \{0, 1\}$. We consider a predictor with logit $f(x, a)$ and logistic loss

$$\ell(f, y) = -y \log \sigma(f) - (1 - y) \log(1 - \sigma(f)), \qquad \sigma(f) = \frac{1}{1 + e^{-f}}.$$

Its empirical risk is

$$\hat{R}_D(f) = \frac{1}{n} \sum_{i=1}^{n} \ell(f(x_i, a_i), y_i).$$

Let $\mathcal{H}_{\text{state}}$ contain all real-valued functions of x alone. This unrestricted class describes what the observed data can distinguish. It is not an assumption about the ability or learned features of a particular model. The logit f is distinct from the video score $s(a)$ in Equation 2. The following results concern a simplified logistic objective, whereas the released models use their original training objectives. They do not establish optimization behavior or held-out generalization.

B.2 Single-Action Observations

Proposition 1. Suppose every observed action satisfies $a_i = \pi(x_i)$ for some function π. For every predictor $f(x, a)$, there is a state-only predictor $h_f(x) = f(x, \pi(x))$ such that

$$\hat{R}_D(h_f) = \hat{R}_D(f).$$

Thus, fitting the observed outcomes does not require distinguishing alternative actions within the same context.

Proof. For every observation, $h_f(x_i) = f(x_i, \pi(x_i)) = f(x_i, a_i)$. The predictors therefore incur identical loss on each observation and have equal empirical risk. □

The result allows an action-insensitive fit. It does not show that a trained network adopts it. Parameter sharing, regularization, and information from related contexts may still support predictions for unobserved actions. Stochastic policy data need not satisfy the single-action assumption, and the proposition does not imply that policy failures are uninformative.

B.3 Matched Actions with Different Outcomes

Proposition 2. Suppose D contains m pairs

$$\big\{(x_j, a_j^+, 1), (x_j, a_j^-, 0)\big\}, \qquad a_j^+ \neq a_j^-,$$

with each observation index used in at most one counted pair. Let I_{rest} contain the remaining observation indices, and define

$$\hat{R}^*_{\text{rest}} = \inf_{h \in \mathcal{H}_{\text{state}}} \frac{1}{n} \sum_{i \in I_{\text{rest}}} \ell(h(x_i), y_i).$$

Then

$$\inf_{h \in \mathcal{H}_{\text{state}}} \hat{R}_D(h) \geq \frac{2m}{n} \log 2 + \hat{R}^*_{\text{rest}}.$$

For an action-conditioned predictor, the summed loss on a pair with logit difference $\Delta = f(x, a^+) - f(x, a^-)$ satisfies

$$\ell(f(x, a^+), 1) + \ell(f(x, a^-), 0) \geq 2 \log\Big(1 + e^{-\Delta/2}\Big).$$

Proof. A state-only predictor assigns the same logit u to both observations in a pair. Their summed loss is

$$\log(1 + e^{-u}) + \log(1 + e^{u}),$$

which is minimized at $u = 0$ with value $2 \log 2$. Since counted pairs do not share observation indices, these bounds can be summed without counting a loss term twice. Adding the lower bound for the remaining observations proves the first inequality.

For a fixed difference Δ, write the two logits as $c + \Delta/2$ and $c - \Delta/2$. Their summed loss is

$$L(c) = \log(1 + e^{-c-\Delta/2}) + \log(1 + e^{c-\Delta/2}).$$

This function is convex and symmetric in c, so its minimum occurs at $c = 0$, giving $2 \log(1+e^{-\Delta/2})$. □

If labels are consistent at each context–action input and the predictor can represent arbitrarily large separating logits, the loss on matched pairs can approach zero. These conditions are not guaranteed for a released model. When counting the m pairs, each observation may be used at most once, even if a failed observation could be paired with several successful observations from the same context.

B.4 Implications for CureWM

The propositions motivate constructing action alternatives with execution-verified different outcomes under matched contexts. They do not establish that successful perturbed replays are necessary for repair: original demonstrations may already provide successful alternatives. Additional successful replays are evaluated empirically rather than justified by a necessity claim. Exact shared contexts are an idealization. A common source demonstration or action prefix does not by itself guarantee identical observed states during replay, especially on hardware. The argument also does not identify shared prefixes as the sole cause of the observed improvements, exclude useful learning from policy failures, or guarantee transfer to new tasks. Training-step comparisons at fixed data composition do not test the bound's dependence on m/n.

C Additional Diagnosis on ALOHA

We evaluate the released Cosmos Policy checkpoint for ALOHA (Zhao et al., 2023) on action sequences from 40 recorded episodes. For each episode, we apply three action edits: freezing the actions, reversing the action sequence, and opening both grippers. The reported changes in predicted task-success values are below 0.01 in magnitude, indicating limited sensitivity to these edits. The modified actions are not re-executed, so their outcomes are not verified. This experiment measures sensitivity to action changes, rather than prediction accuracy on confirmed failures. ALOHA is included for diagnosis only and we do not evaluate CureWM repair on this checkpoint.

D LIBERO

D.1 Training Data and Evaluation Splits

The official training data contain 2,000 records, including 300 failed policy rollouts. CureWM adds 822 counterfactual replay records: 242 verified failures and 580 successful replays. These include twelve repeated successful records. The failures comprise 23 approach-overshoot, seven carry-slip, 32 contact-oscillation, 56 insufficient-grip, 67 premature-release, and 57 wrist-tilt replays. Using one-based task and demonstration indices, counterfactual training uses demonstrations 1–3 per LIBERO-Goal task. The first evaluation set uses demonstrations 4–6, the second uses 7–9, and the other-demo comparison uses demonstrations 10–12. Sources whose original action sequence fails during replay are skipped, leaving 27 training sources and 56 evaluation sources. The second evaluation set is generated after the compared models finish training. The evaluation contains 484 failures from 45 source demonstrations and 1,196 successful perturbed replays. All compared models use the same evaluation records. The training-pool evaluations instead use two groups containing 115 and 127 failures. These are in-sample for CureWM and do not establish held-out repair. Most evaluation demonstrations also occur in the shared official training data. Three are absent: task 4/demonstration 6, task 7/demonstration 5, and task 4/demonstration 9. They contribute 20, 12, and 21 failures, respectively. On these 53 failures, value optimism is 26.4%, 28.3%, and 11.3% for the released model, baseline, and CureWM. With only three source demonstrations, this subset does not isolate demonstration familiarity from task and failure-type differences. The LIBERO-Spatial, LIBERO-Object, and LIBERO-10 evaluations contain 367/443, 286/524, and 359/331 failed/successful perturbed replays, respectively. Their official data remain included in fine-tuning, but their counterfactual replays do not enter training. The suite means in Table 1 weight the four suites equally.

D.2 Training Configuration and Checkpoints

Cosmos Policy fine-tunes the full prediction network, rather than a selected head or adapter. The configuration sets `demonstration_sampling_prob=0.5` and `success_rollout_sampling_prob=0.5`. Successful and failing trajectories are sampled through the existing training procedure, not supplied as explicit positive–negative pairs. Training-record counts therefore do not directly specify sampling probabilities. LIBERO uses batch size 4 and a default learning rate of 10^{-5}. Primary comparisons use 17,500 steps because this is the checkpoint retained for the primary CureWM model within a 20,000-step schedule. This evaluation step was not a prospectively specified stopping rule. Table 6 reports the retained checkpoints and corresponding results.

Table 6: Four independently fine-tuned CureWM models on LIBERO. Values are optimism percentages. Training groups A and B contain 115 and 127 failures, while held-out evaluation uses the same 484 failures for every model.

Model	Steps	Training A	Training B	Held-out
Primary	17,500	16.5	17.3	30.2
Supplementary 1	20,000	28.7	29.9	43.0
Supplementary 2	20,000	18.3	20.5	37.6
Supplementary 3	17,500	21.7	32.3	40.9

LIBERO retains the native discounted-return supervision with $\gamma = 0.99$. For each action pair, the probe selects the frame immediately before the first action difference, adjusting the index when necessary to retain a full 16-action chunk. Both predictions use the same observation, task instruction, query time, and generation seed. The probe generates a future state under each chunk and then queries its value. The resulting value is not treated as a calibrated probability of eventual task success.

The first model is designated primary and also achieves the lowest observed held-out optimism. The 17,500-step checkpoints for supplementary models 1 and 2 were not retained. Variation therefore combines training randomness and training duration. All four models are compared with the same primary baseline, whose optimism is 79.5%; a second official-data baseline achieves 78.5%. The comparisons do not use four independently seed-matched baselines. At learning rate 5×10^{-6}, three

additional models reduce the training-pool optimism ranges from 12.2 to 4.4 percentage points on group A and from 15.0 to 5.5 on group B. Mean optimism is approximately 2.5 points higher. These are training-pool results, not additional held-out repair estimates. Closed-loop evaluation uses the released evaluation script, action chunks of length 16, five action denoising steps, and evaluation seed 195. Each suite contains ten tasks with 50 episodes per task, giving 500 episodes per model and suite.

For the other model settings, RoboCasa uses batch size 2 and three camera views. The Ctrl-World protocols use 1,400 adaptation steps followed by 3,600 comparison steps. Additional hardware adaptation and cube-task evaluations are specified in Appendices G.2 and G.4. The submitted configurations and analysis scripts provide the executable settings.

D.3 Discrimination within Perturbation Types

We evaluate the same 484 failures and 1,196 successful perturbed replays. Table 7 separates results by perturbation type. Improvement is concentrated in insufficient grip, premature release, and wrist tilt. Wrist tilt also produces the largest increase in successful replays incorrectly predicted to fail.

Table 7: LIBERO-Goal discrimination by perturbation type. AUROC and false-positive rates compare the fine-tuning baseline with CureWM on identical replays. False-positive rates are percentages of successful replays assigned a value at or below 0.5.

			AUROC		False-positive rate	
Perturbation	Failed	Successful	Baseline	CureWM	Baseline	CureWM
Approach overshoot	33	247	0.88	0.87	14.6	19.0
Carry slip	12	268	0.68	0.70	27.6	29.5
Contact oscillation	74	206	0.52	0.51	34.5	37.9
Insufficient grip	122	158	0.45	0.86	43.7	44.3
Premature release	140	140	0.57	0.78	42.1	42.9
Wrist tilt	103	177	0.41	0.72	35.0	68.4

To control for severity as well as perturbation type, we compute AUROC separately within each combination. The summary equally weights the 21 groups containing at least three successful and three failing replays. It is 0.537 for both the released model and baseline, and 0.714 for CureWM. This differs from pooled AUROC and from an average over perturbation types alone. Table 8 reports AUROC at fixed perturbation type and severity

Failure counts at severities 0.4, 0.6, 0.8, and 1.0 are $(7, 6, 10, 8)$ for approach overshoot, $(0, 2, 4, 5)$ for carry slip, $(8, 13, 21, 30)$ for contact oscillation, $(0, 40, 41, 40)$ for insufficient grip, $(21, 39, 39, 40)$ for premature release, and $(12, 23, 31, 35)$ for wrist tilt. The remaining replays in each group succeed. Outcome labels and evaluation records are shared across models.

D.4 Leave-One-Perturbation-Type-Out Evaluation

For each evaluated perturbation type, we remove all its counterfactual training replays while retaining official data and the other types. Two independently fine-tuned models are evaluated at 17,500 steps on failures of the excluded type from training-pool groups A and B. These excluded replays do not enter that model's counterfactual training, but other replays from the same source demonstrations can remain. The experiment tests transfer across perturbation types, not unseen source demonstrations. Table 9 reports the results for each excluded perturbation type.

Insufficient grip and premature release show partial transfer. Contact oscillation is unchanged on B, while A contains only 13 failures. Wrist tilt improves slightly on A but worsens on B. Carry slip has only one A failure and six B failures. Approach overshoot is not evaluated in this experiment.

D.5 Failure-Source and Label Comparisons

Fresh policy failures. We execute the released policy with stochastic sampling for 16 passes over ten LIBERO-Goal tasks and 50 initial states per task, totaling 8,000 episodes. The collection yields

Table 8: AUROC at fixed perturbation type and severity. Each group contains 56 replays. Dashes indicate no failures. Severity 0.2 is omitted because every type contains at most two failures at that severity. Carry slip at severity 0.6 contains only two failures and is excluded from the 21-group mean. Means use values before rounding.

Perturbation	Model	0.4	0.6	0.8	1.0
Approach overshoot	Released	0.85	0.94	0.89	0.96
	CureWM	0.85	0.95	0.88	0.95
Carry slip	Released	—	0.55	0.72	0.57
	CureWM	—	0.56	0.75	0.57
Contact oscillation	Released	0.67	0.52	0.55	0.38
	CureWM	0.66	0.52	0.56	0.39
Insufficient grip	Released	—	0.36	0.29	0.36
	CureWM	—	0.58	0.88	0.93
Premature release	Released	0.55	0.38	0.37	0.36
	CureWM	0.56	0.49	0.84	0.90
Wrist tilt	Released	0.44	0.42	0.37	0.31
	CureWM	0.61	0.66	0.69	0.78
Mean over 21 groups	Released		0.537		
	Baseline		0.537		
	CureWM		0.714		

Table 9: Value optimism after excluding a perturbation type from counterfactual training. Each result lists the released model followed by two independently fine-tuned models.

Excluded type	n_A	n_B	A: Released / models 1, 2	B: Released / models 1, 2
Carry slip	1	6	100 / 0, 0	16.7 / 16.7, 16.7
Contact oscillation	13	19	92.3 / 53.8, 69.2	57.9 / 57.9, 57.9
Insufficient grip	31	25	80.6 / 45.2, 19.4	60.0 / 24.0, 40.0
Premature release	36	31	88.9 / 41.7, 55.6	51.6 / 22.6, 22.6
Wrist tilt	24	33	83.3 / 75.0, 75.0	57.6 / 69.7, 69.7

238 verified failures. 61.8% occur on two tasks, while one task has no observed failure. Zero observed failures does not imply zero failure probability.

For each task, we take the smaller of the available policy-failure and counterfactual-failure counts. This gives 140 distinct failures per source. Each set is repeated to form 242 training records and combined with the same 580 successful counterfactual replays and 2,000 official records. Both models train for 7,500 steps. Thus distinct-failure counts, per-task counts, training-record counts, successful replay data, and training duration are matched.

Of the 140 selected policy failures, 69 have a recorded success from the same task and initial-state index. None shares an exactly identical first executed action with its successful counterpart. This criterion requires numerical equality of every action component but it does not test approximate action similarity. All collected policy failures reach the 300-step limit, which does not identify when a failure-causing event occurred. These construction differences do not establish a unique explanation for the matched comparison in Table 5.

Existing policy failures. The own-failure comparison repeatedly samples 300 failures already present in the official mixture to produce 822 additional training records. Unlike the fresh comparison, it does not match task coverage, novelty, or repetition. It therefore tests this specific reuse of existing failures rather than policy-failure learning in general.

Successful replay partners. No-graded removes the 580 successful replay records. Other-demo replaces them with 580 verified successes from demonstrations 10–12 per task, approximately matched by perturbation type and severity. Both retain the 242 verified failures and official training data. The full other-demo training list contains 2,822 records. These comparisons change training-data composition; record counts alone do not guarantee equal exposure under the sampling procedure.

Outcome verification. The unverified-label comparison assigns failure labels to all perturbed replays, including the 580 successful records, while retaining the remaining training procedure. It achieves 57.9% held-out optimism and a value gap of 0.103, compared with CureWM's 30.2% and 0.282.

On training failures, its optimism is 10.3%, below CureWM's 16.9%. However, its mean value for successful actions is only 0.147 and its value gap is 0.062. Lower training optimism therefore overstates repair: the model also lowers successful-action predictions.

Failed-action supervision. Masking only action-prediction loss on failed LIBERO replays gives 32.9% held-out optimism and a value gap of 0.258, compared with 30.2% and 0.282 without masking. Substantial value-side repair remains without supervising those failed actions. This prediction comparison does not establish a closed-loop benefit from masking.

D.6 Value and Video Predictions

Cosmos Policy's value and video predictions do not improve equally. On the first held-out set of 231 failures, value optimism is 71.4%, 72.7%, and 28.1% for the released model, baseline, and CureWM, while visual optimism is 58.0%, 55.8%, and 45.0%.

Wrist tilt illustrates the remaining visual error: among its 49 failures, baseline-to-CureWM value optimism falls from 93.9% to 24.5%, but visual optimism falls only from 100.0% to 81.6%. The results support partial video repair, rather than uniform improvement across prediction outputs.

E RoboCasa

E.1 Training Data and Evaluation Sets

The initial grasp-focused pool (the Grasp replays setting in Table 2) contains 1,291 official rollouts and 513 counterfactual replays, including 262 verified failures. The expanded pool adds 456 drawer and door replays, including 113 failures. This changes both task coverage and data quantity.

We distinguish three prediction evaluations:

1. The 120 modified-action examples measure Δ_{SF} on states from official rollouts. Modified actions are not re-executed. All value gaps in Table 2 use this same evaluation.
2. The official-rollout evaluation contains 148 successes and 92 failures included in the training mixture. It provides supporting results, not held-out evidence.
3. The separate drawer and door evaluation contains 111 successes and 114 failures excluded from training. All held-out AUROCs in Table 2 use this set at 75% of episode length.

Robot control is evaluated over three evaluation seeds with ten episodes each, giving 30 episodes per task and model. Neither OpenDrawer nor CloseDoubleDoor is covered by the initial counterfactual training pool.

E.2 Training Duration and Action Supervision

Table 2 reports the primary comparisons. At 20,000 steps, a second independently fine-tuned model using the initial pool gives $\Delta_{\mathrm{SF}} = 0.0357$, compared with 0.0342 for the primary model. Two models trained by up-sampling existing policy failures give 0.0131 and 0.0274.

The action-loss comparison masks only action-frame loss on failed rollouts, retaining their value and future-state supervision. The pool, initialization, training seed, batch size, and learning-rate schedule remain fixed. At 5,000 steps, this improves observed task success but reduces the value gap from 0.0672 to 0.0308. It supports failed-action supervision as one contributor to control-performance loss without identifying all contributing mechanisms.

The recorded paired comparison on 120 modified-action examples gives $t = 10.39$, $p < .001$, and $d_z = 0.948$ for unmasked treatment relative to the released model, with 109 positive differences. Masked treatment gives $t = 8.5$ and $d_z = 0.78$, with 100 positive differences. The masked-minus-unmasked comparison gives $t = -17.4$. These example-level tests do not account for dependence among examples from the same trajectory and do not test differences in robot task success.

E.3 Additional Prediction and Control Results

On the official-rollout evaluation, predictions are assessed at 25%, 50%, and 75% of episode length. At 75%, recall at a 10% false-positive rate increases from 53% to 70%. These rollouts occur in the training mixture, so this result does not establish generalization to unseen rollouts or a deployment operating point.

On coffee-mug manipulation, absent from the initial counterfactual pool, the value gap changes from -0.006 to $+0.058$ at 5,000 steps. This measures sensitivity to modified actions, not verified failure detection. Value-guided best-of-N selection provides no statistically established improvement in robot task success at the evaluated scale.

F Ctrl-World

F.1 Video Evaluation and Data Splits

The simulator evaluation uses the frozen SVD VAE representation and three camera views. Distances average squared differences over the representation dimensions of all views. Predictions are evaluated after four chained interaction steps, corresponding to 16 raw frames after the initial context.

The expanded evaluation contains 30 action pairs from 14 source demonstrations. Ten pairs from nine sources are excluded from counterfactual training. The full set partially overlaps training and is reported separately from this held-out subset in Table 3.

Positive $s(a^-)$ indicates a prediction closer to the recorded success than the recorded failure. A reduction in mean $s(a^-)$ is not by itself proof of more accurate generated video. The diagnostic requires recorded outcome references.

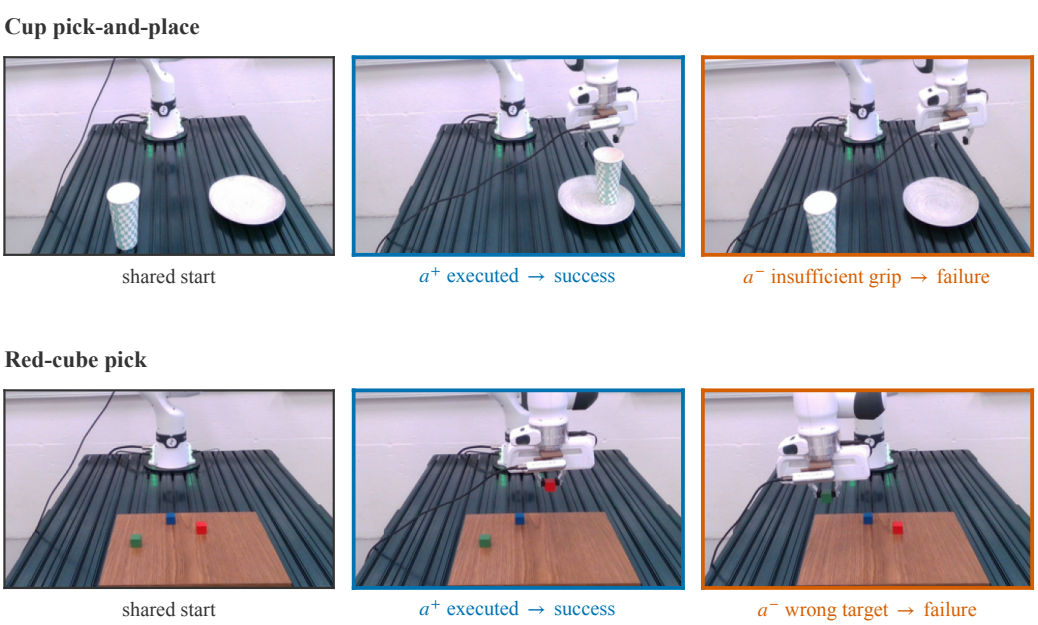


Figure 4: **Hardware tasks and executed outcomes.** Each row shows the starting scene, a successful execution, and a failing execution. Top: cup pick-and-place, with failure caused by insufficient grip. Bottom: red-cube picking, with failure caused by grasping a distractor instead of the target cube. All images show physical executions, not model predictions.

F.2 Paired Comparisons and Independent Training

Paired comparisons evaluate the baseline and CureWM on identical examples. The fixed analysis in `cw_paired_ci.py` resamples source demonstrations with replacement, retains all pairs from each sampled source, and recomputes the mean using the resulting number of pairs. It uses seed 0 and 20,000 resamples.

The mean baseline-minus-CureWM failure-score difference is 0.053 on all 30 pairs and 0.066 on the ten held-out pairs. On the held-out subset, three optimistic predictions are corrected and one changes in the opposite direction. The paired analysis supports a score decrease, while the thresholded optimism difference is not statistically established.

A second independently fine-tuned CureWM model gives a smaller full-set score improvement of 0.0144, with reported $p \approx .18$. Its mean held-out failure-action score is 0.0641. The independently trained models are not pooled as repeated observations of the same model.

F.3 Original 23-Pair Evaluation

The original evaluation contains 23 pairs and partially overlaps counterfactual training. Visual optimism is 95.7%, 91.3%, and 60.9% for the adapted model, baseline, and CureWM. Relative to the baseline, seven predictions are corrected and none reversed, giving a two-sided exact McNemar $p = 0.015625$. The optimism difference is -30.4 percentage points.

The second CureWM model's reported comparison on this original set has $p \approx .026$. This is a different evaluation from the expanded 30-pair set, on which its improvement is not statistically established. Results from the two sets are not interchangeable.

G Real Robot

Figure 4 illustrates the two hardware tasks and their execution-verified successful and failing outcomes.

G.1 Data Collection and Splits

Forty scripted cup pick-and-place demonstrations produce 140 perturbed replays at severities $\{0.6, 0.8, 1.0\}$: 107 verified failures and 33 successful replays. The perturbation types are insufficient grip, premature release, carry slip, and wrist tilt. Only wrist tilt produces successful perturbed replays at the tested severities.

Ten demonstrations and their 20 replays are reserved for evaluation before training. These replays contain 15 failures and five successes. Neither the demonstrations nor their replays enter training. The complete split is shown in Table 10.

Table 10: Cup-task data split. The 31 unperturbed training trajectories contain 28 demonstrations and three additional replays. Successful perturbed replays all use wrist tilt.

Data	Training	Validation	Evaluation
Failed perturbed replays	90	2	15
Successful perturbed replays	24	4	5
Unperturbed successful trajectories	31	2	10

After 1,400 steps of success-only scene adaptation, CureWM and the baseline receive 3,600 further training steps. CureWM uses all 145 training trajectories; the baseline uses the 31 unperturbed trajectories. Video losses have equal weight across training records.

The 71-failure diagnostic evaluates the adapted model before repair. Most examples subsequently enter repair training, so the diagnostic is distinct from the held-out repair evaluations.

G.2 Prediction Scores and Scene Adaptation

Camera inputs and score definitions. The hardware has external and wrist cameras, but the model receives three inputs: external, external, and wrist. Distance calculations average over all three, giving the external view twice the weight of the wrist view.

Both action scores use Equation 2. Visual optimism counts $s(a^-) > 0$; pair-ranking accuracy counts $s(a^+) > s(a^-)$; and Δ_{SF} averages their difference. In the recorded outputs, `fidelity_plus` is distance to the successful reference, not the signed score:

$$s(a^+) = \texttt{fidelity_plus_cross} - \texttt{fidelity_plus}.$$

The two-reference failure detector uses $-s(a)$. The success-reference-only detector uses $d(\hat{z}_T(a), \phi(y_T^+))$. Both require recorded reference outcomes. Across the two evaluations, baseline-to-CureWM AUROC is 0.66 to 0.96 with both references, and 0.67 to 0.90 with the successful reference only. Table 4 reports two-reference AUROC; the main-text 0.67-to-0.90 result uses one reference.

Rendering-quality screening. For evaluation pairs $\mathcal{P}$, we compute

$$\mathrm{SNR} = \frac{|\mathcal{P}|^{-1} \sum_{i \in \mathcal{P}} d(\phi(y_{T,i}^+), \phi(y_{T,i}^-))}{|\mathcal{P}|^{-1} \sum_{i \in \mathcal{P}} d(\hat{z}_{T,i}(a_i^+), \phi(y_{T,i}^+))}. \tag{6}$$

The heuristic requires $\mathrm{SNR} > 1$. It compares separation between recorded outcomes with prediction error on successful actions and is a ratio of mean distances. Passing does not guarantee valid inference.

A re-check of an earlier sponge evaluation found SNR values of 0.73 and 0.52 and led to withdrawal of its transfer interpretation. The rule was subsequently fixed before the cube-task and second cup-task evaluations, not before the original sponge experiment.

Second hardware evaluation. Twenty new demonstrations produce 40 replays, with two perturbation types per demonstration at severity 1.0. Insufficient grip, premature release, and carry slip each fail on 10/10 replays; wrist tilt succeeds on 10/10.

Before additional adaptation, SNR is 0.85/0.83/1.02/0.97 for the adapted model, baseline, and CureWM models 1 and 2. Three models fail the screen. Their optimism rates are 90/90/10/17%; these measurements are excluded from the combined repair results.

Ten demonstrations are then used for additional scene adaptation and ten for evaluation, balanced over start locations and perturbation types. All models receive 350 steps on the same successful demonstrations with unchanged normalization statistics. Evaluation demonstrations and replays remain excluded. The expected repair advantage and requirement to pass the screen are recorded before this training.

After adaptation, SNR is 1.11/1.15/1.24/1.14. Results on the 15 held-out failures appear in Table 4. Combining this evaluation with the first gives 30 pairs from 20 demonstrations. Each model uses the adaptation corresponding to its evaluation.

Successful-replay comparison. The additional baseline combines 31 unperturbed trajectories with the same 24 successful wrist-tilt replays used by CureWM. Initialization and 3,600 training steps are matched. First-evaluation optimism remains 15/15, while mean $s(a^-)$ decreases from 0.184 to 0.138. This does not reproduce CureWM's repair. However, 55 versus 145 distinct trajectories remain unmatched, so data content, quantity, and diversity are not fully separated.

G.3 Human Assessment

Predictions for both actions in the first evaluation's 15 held-out pairs are generated by the adapted released model and CureWM, producing 60 clips. Each rater sees all 30 failing-action clips and ten successful-action clips, five per model. Three non-author volunteers unfamiliar with the project classify clips as successful, failed, or unclear. Raters were informed of the procedure and research use of their ratings and voluntarily agreed to participate. Only generated videos are shown, without recorded reference outcomes. Clip identifiers are hashed, order is independently shuffled for each rater, and the interface does not load the answer key. All raters complete 40 clips and none selects unclear. The fine-tuning baseline is not evaluated. Table 11 summarizes the majority judgments.

Table 11: Majority judgments from three raters. The diagnostic classifies predictions as success-like when $s(a) > 0$.

Executed action	Model	Human: successful	Diagnostic: success-like
Failing	Adapted	8/15	13/15
Failing	CureWM	0/15	6/15
Successful	Adapted	3/5	—
Successful	CureWM	5/5	—

Individual raters classify 8, 7, and 8 adapted-model failing-action predictions as successful, versus zero for CureWM. Pairwise Cohen's κ values are 0.95, 0.90, and 0.84. Human judgments agree on the direction of improvement, but the distance diagnostic more frequently classifies predictions as successful. This small evaluation does not calibrate the separate 71-failure diagnostic or establish preservation of successful predictions.

G.4 Cube Picking and Cross-Task Transfer

The robot picks a colored cube among distractors. Collection contains 112 episodes across 29 scenes, including 83 verified failures. An operator places objects between scenes, and scripts execute the trajectories. Gripper telemetry distinguishes a held 2 cm cube (approximately 0.70) from an empty close (approximately 1.00), using threshold 0.88. For `wrong_target`, target identity additionally uses known placement geometry and the commanded distractor target, with operator review of anomalies. Ten scenes containing 29 counterfactual pairs are reserved for evaluation and excluded from cube-scene adaptation and repair. Split membership is derived from the training split file. The baseline and task-specific CureWM start from the same cube-adapted checkpoint. An initial evaluation at SNR 0.84 is excluded; additional scene adaptation raises SNR to 1.17. We evaluate both models at 300, 600, 1,200, 1,800, 2,400, 3,000, and 3,600 training steps, giving fourteen

model–checkpoint evaluations. The primary comparison uses 3,600 steps. All fourteen model–checkpoint evaluations pass the screen, with SNR 1.06–1.18. Table 12 compares task-specific repair with transfer from cup-task repair.

Table 12: Cube picking on 29 held-out pairs from ten scenes. Differences are baseline minus repaired-model mean $s(a^-)$; positive values indicate lower failure-action scores after repair.

Repair setting	Failure-score difference
Cube-task repair	$+0.025$
Cup-task repair before cube adaptation	-0.007

Across checkpoints from 600 steps onward, detection AUROC ranges from 0.927 to 0.976 for CureWM and from 0.748 to 0.924 for the baseline. These are changes along training trajectories, not variation across independently trained models. The cup-to-cube comparison does not establish beneficial transfer; the positive result uses cube-specific counterfactual training data.