# CORRELATION DOES NOT CERTIFY RELIABLE EVALUATION: IDENTIFIABILITY LIMITS OF WORLD-MODEL-BASED ROBOT POLICY SELECTION

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Policy evaluation with world-model rollouts and a VLM judge has been reported to correlate with measured success rates. We ask whether this correlation certifies sensitivity to action consequences. Empirically, it does not. An evaluator whose scores correlate with simulator success rates at $r = 0.72$ changes its success judgment in only $0.06$ of applicable episodes under an intervention that makes every modified action sequence fail. A grasp-delay perturbation likewise lowers policy success without lowering the evaluator score, while evaluator scores are strongly associated with distance to the training demonstrations. We prove that, without additional real-world trials, no evaluator can reliably rank two candidates whose difference depends on an action unsupported by the demonstrations, even given a pretrained world model, the candidates' weights, and a perfect judge. Identifiability depends on expected demonstration support, and perfect rank correlation can coexist with intervention insensitivity. A controlled learned-world-model experiment confirms this dependence. Finally, we study ResGap, an empirical coarse-screening heuristic that requires neither world-model rollouts nor a VLM judge, as one possible offline stage before target-environment evaluation.

## 1 INTRODUCTION

Developing robot manipulation policies from demonstrations involves selecting among many candidates: architectures, training configurations, and checkpoints. Estimating each candidate's success rate requires many real-world trials. Real-world evaluation is therefore a major bottleneck in policy development (Li et al., 2025a; Zhou et al., 2025).

World-model-based policy evaluation has been proposed to relieve this bottleneck. A candidate policy is rolled out in an action-conditioned video model, and a vision-language model (VLM) judges the generated rollout to estimate its success rate without real-world trials. Several such evaluators have been proposed (Li et al., 2025b; Quevedo et al., 2026; Team et al., 2025; Li et al., 2026), and are typically validated by correlation or rank agreement between estimated and measured success rates.

Prior work has observed that a rollout can appear successful even when the corresponding action fails in the target environment. Such failures have been attributed to the world model, motivating architectures or training objectives that improve action controllability (Li et al., 2026; Jeon et al., 2026). We instead ask whether correlation with measured success rates is sufficient to certify sensitivity to action consequences. We show that when the action distinguishing two candidates is unsupported by the demonstrations, no evaluator can reliably rank them without additional real-world trials.

We audit such an evaluator on LIBERO using 31 candidates from five policy families. Among the 24 ACT and Diffusion Policy candidates that share the world model's action interface, its score correlates with simulator success rate. However, the evaluator barely responds to an intervention that makes the modified action sequences fail. Across all 31 candidates, the evaluator score is strongly associated with distance to the training demonstrations. A grasp-delay perturbation further reduces success across five ACT checkpoints without reducing the evaluator score. These results show that correlation with success can coexist with weak sensitivity to action consequences.



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.

We formalize this limitation independently of the particular world model or judge. Without additional real-world trials, no evaluator can reliably identify the better of two candidates when their difference depends on an action unsupported by the demonstrations. This holds even given a pretrained world model, the candidates' weights, and a perfect judge. Identifiability depends on the expected demonstration support for that action rather than on the total number of demonstrations. Moreover, perfect rank correlation over a candidate family can coexist with complete insensitivity to interventions on that action.

We next test this support dependence with a learned world model in a controlled environment. When $\varepsilon = 0$, increasing the number of demonstrations does not reduce error on the unsupported action consequence. For $\varepsilon > 0$, the error instead decreases with $n\varepsilon$. Held-out prediction loss remains similar across conditions and does not reveal the missing action consequence.

The theorem also suggests a narrower role for evaluation without real-world trials: screening candidates for consistency with demonstrated behavior while leaving unsupported differences to real-world evaluation. We instantiate this idea with ResGap, an empirical coarse screening heuristic that requires neither world-model rollouts nor a VLM judge. Under leave-one-training-series-out evaluation on 17 ACT checkpoints, ResGap attains higher point estimates for rank correlation and lower MAE than the world-model evaluator, but is sensitive to the demonstration-pool construction. We therefore treat ResGap only as a coarse screening heuristic before target-environment evaluation.

## 2 Related Work

**World-model-based policy evaluation.** Recent methods evaluate candidate policies by rolling them out in action-conditioned video models and comparing the resulting scores with measured success rates (Li et al., 2025b; Quevedo et al., 2026; Tseng et al., 2025; Team et al., 2025; Li et al., 2026). General-purpose robot world models have also adopted policy evaluation as an application (Zhu et al., 2025; Guo et al., 2026; Huang et al., 2026; Wang et al., 2026; Gao et al., 2026). Existing methods are typically validated by correlation or rank agreement with measured performance, while several studies have reported generated rollouts that appear successful despite actions that fail in the real environment (Li et al., 2026; Jeon et al., 2026). Model-level diagnostics can detect whether generated videos follow commanded actions (Tseng et al., 2026), but they do not determine the real outcome of an unsupported action. Proposed remedies otherwise focus on improving action controllability in the world model (Li et al., 2026; Jeon et al., 2026; Tseng et al., 2026). We instead study the limitation of evaluation without additional real-world trials and show that unsupported action differences are not identifiable from the available demonstrations alone.

**Label-free performance estimation.** Related problems arise in label-free performance estimation, where model accuracy on unlabeled target data is predicted from distribution statistics, relations between in- and out-of-distribution accuracy, model confidence, or agreement between models (Deng & Zheng, 2021; Miller et al., 2021; Garg et al., 2022; Baek et al., 2022). Without assumptions on the distribution shift, however, target accuracy cannot in general be identified reliably (Garg et al., 2022). Our setting is more restrictive in a different way: the estimator without real-world trials observes neither the states visited by a candidate nor their outcomes.

**Lower bounds for offline evaluation and selection.** Selecting among candidates from logged demonstrations is closely related to offline policy selection (OPS), which is typically approached through off-policy evaluation (OPE) or value-function criteria (Jiang & Li, 2016; Zhang & Jiang, 2021; Tang & Wiens, 2021). OPS is in the worst case as hard as OPE (Liu et al., 2023b), and existing lower bounds show that data coverage is necessary for reliable offline evaluation and optimization (Chen & Jiang, 2019; Wang et al., 2020; Zanette, 2021; Foster et al., 2022). Our Theorem 1 uses two-environment indistinguishability argument (Tsybakov, 2009), but under a stronger access model: the evaluator may use a pretrained world model, the candidates' weights, and the exact success function. Because these objects are fixed before observing the environment, the evaluator still obtains environment-specific information only through the demonstrations.

**Reducing real-world trials.** Evaluation without real-world trials can still be used to screen candidates before real-world testing. ResGap is related to ProjNorm, which predicts out-of-distribution error from the parameter change induced by fine-tuning on pseudo-labeled target data (Yu et al.,

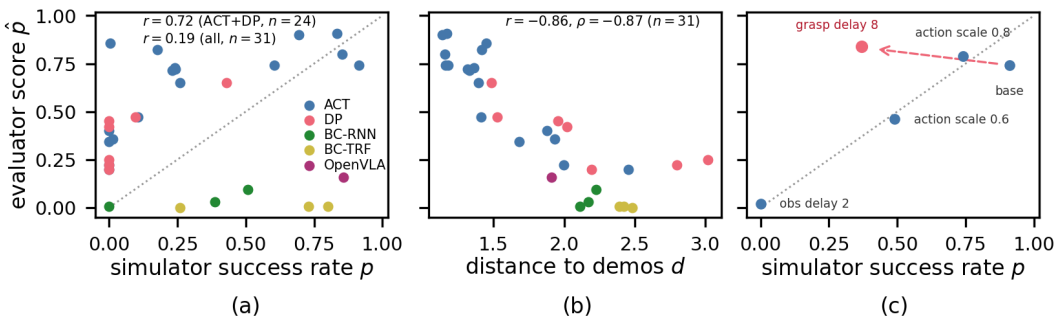


Figure 1: Auditing the world-model evaluator. (a) Evaluator score $\hat{p}$ versus simulator success rate $p$ for 31 candidates (24 ACT+Diffusion Policy: $r = 0.72$; all: $r = 0.19$). (b) $\hat{p}$ versus distance $d$ to the training demonstrations ($r = -0.86$). (c) Test-time perturbations of an ACT checkpoint; grasp delay lowers $p$ from $0.91$ to $0.37$ while raising $\hat{p}$ from $0.74$ to $0.84$.

2022); ResGap instead uses demonstrated actions and compares parameter changes on real and world-model-regenerated demonstrations. Real-to-sim evaluation reduces physical testing by evaluating policies in a simulator matched to the real setup (Li et al., 2025a), while automated and distributed evaluation reduce the labor or distribute the cost of real-world trials (Zhou et al., 2025; Atreya et al., 2025). Sequential tests reduce the number of trials needed for policy comparison (Snyder et al., 2025). SureSim combines simulator rollouts with a small number of real trials to estimate real success rates (Badithela et al., 2025), whereas ResGap is used only for coarse screening before target-environment evaluation.

## 3 Correlation Without Sensitivity: Auditing a World-Model Evaluator

In this section, we audit whether a world-model evaluator that correlates with measured success rates is sensitive to the consequences of actions. We first decompose its evaluation error, then intervene on the action sequence and test whether the generated outcome changes accordingly. Finally, we examine how the evaluator score relates to the distance of generated rollouts from the training demonstrations.

### 3.1 Evaluation Pipeline and Setup

A world-model evaluator runs a candidate policy in closed loop inside an action-conditioned video model and uses a VLM judge to label each generated episode as success or failure. We denote by $\hat{p}$ the fraction of generated episodes that the VLM judge labels as successful, and the success rate of the same policy in the target environment by $p$. The evaluator uses $\hat{p}$ as an estimate of $p$ without executing the policy in the target environment.

Our evaluator follows the closed-loop configuration of WorldGym (Quevedo et al., 2026). We fine-tune Wan2.1-1.3B (Wan et al., 2025) on 450 LIBERO-Object demonstrations (Liu et al., 2023a) and use Qwen2.5-VL (Team, 2025) as the judge, with majority voting over three prompts. We use the LIBERO simulator as the target environment because it provides both $p$ and the outcome of arbitrary modified action sequences. Therefore, each world-model rollout starts from the initial frame of the corresponding simulator episode.

We evaluate 31 candidates from five policy families: ACT (17) (Zhao et al., 2023), Diffusion Policy (7) (Chi et al., 2025), BC-RNN (3) and BC-Transformer (3) (Mandlekar et al., 2022), and OpenVLA (1) (Kim et al., 2025). ACT and Diffusion Policy share the chunked action interface used by the world model. Across these 24 candidates, $\hat{p}$ and $p$ have a Pearson correlation of $0.72$ and a Spearman correlation of $0.82$, consistent with the validation criterion used in prior work. Across all 31 candidates, however, the correlation drops to $0.19$ and $0.23$, respectively (Figure 1(a)).

Table 1: Error decomposition of Eq. (1) by policy family. Each entry is the corresponding term averaged over the candidates of the family, so the three entries in a row sum to the mean error $\hat{p} - p$. A positive entry raises $\hat{p}$ above $p$.

| Family | Candidates | Behavior change | Judge error | Consequence error |
|---|---|---|---|---|
| ACT | 17 | $-0.16$ | $+0.28$ | $+0.20$ |
| Diffusion Policy | 7 | $-0.06$ | $+0.28$ | $+0.09$ |
| BC-RNN | 3 | $-0.29$ | $+0.04$ | $-0.00$ |
| BC-Transformer | 3 | $-0.60$ | $+0.02$ | $-0.02$ |
| OpenVLA | 1 | $-0.76$ | $+0.14$ | $-0.08$ |
| All | 31 | $-0.21$ | $+0.23$ | $+0.19$ |

### 3.2 Error Decomposition

Even when $\hat{p}$ correlates with $p$, their difference can arise from three sources: (i) the policy may behave differently inside the world model because it observes generated rather than target-environment observations; (ii) the judge may misclassify an episode; and (iii) the generated video may depict an outcome inconsistent with the executed actions. Let $p_{\mathrm{cf}}$ denote the success rate obtained by re-executing the action sequences produced inside the world model in the simulator, and let $j_{\mathrm{cf}}$ denote the fraction of these re-executed episodes that the same judge labels as successful. Then:

$$\hat{p} - p = \underbrace{\left(p_{\mathrm{cf}} - p\right)}_{\text{behavior change}} + \underbrace{\left(j_{\mathrm{cf}} - p_{\mathrm{cf}}\right)}_{\text{judge error}} + \underbrace{\left(\hat{p} - j_{\mathrm{cf}}\right)}_{\text{consequence error}} . \tag{1}$$

Because the three terms can cancel, we analyze them separately. In particular, $\hat{p} - j_{\mathrm{cf}}$ compares the generated and re-executed videos for the same action sequence under the same judge, isolating consequence error from judge error. Table 1 reports the three terms averaged within each policy family.

For BC-RNN, BC-Transformer, and OpenVLA, which do not share the chunked action interface, behavior change is large and negative: $-0.29$, $-0.60$, and $-0.76$, respectively, while the other two terms are at most $0.14$ in absolute value. Thus, the action sequences these policies produce inside the world model succeed substantially less often when re-executed in the simulator than the policies themselves do in the simulator. This behavior change helps explain why the correlation drops from $0.72$ among the 24 ACT and Diffusion Policy candidates to $0.19$ across all 31 candidates.

For the 17 ACT candidates, behavior change is $-0.16$, judge error $+0.28$, and consequence error $+0.20$. Across the 7 Diffusion Policy candidates, the corresponding terms are $-0.06$, $+0.28$, and $+0.09$. The smaller behavior-change term for Diffusion Policy should not be interpreted as stronger transfer: five of the seven candidates have $p = p_{\mathrm{cf}} = 0$, making this term zero for those candidates. For these low-performing Diffusion Policy candidates, the discrepancy between $\hat{p}$ and $p$ arises mainly from judge error, whereas ACT exhibits both judge and consequence errors. Thus, the dominant error source varies across candidate regimes rather than following a single pattern.

### 3.3 Intervention on the Action Sequence

Generated videos are judged successful more often than the corresponding re-executed episodes, but this discrepancy does not establish whether the world model responds to changes in the action sequence. A positive consequence error could arise simply because ambiguous outcomes are rendered as successes. Therefore, we take episodes that the evaluator judged successful, insert a grasp release during transport, and regenerate each video from the modified action sequence. For the success-to-failure analysis, we focus on episodes whose original action sequences succeed when re-executed in the simulator. If the world model follows the modified actions, the generated video should show the resulting failure and the judgment should change accordingly.

Re-executing the modified sequences in the simulator causes every episode to fail, yet the VLM judgment changes from success to failure in only $0.06$ of the episodes. This is not explained by the judge alone: the generated videos themselves barely change, with a mean absolute pixel difference of $0.54$ on the 0–255 intensity scale. Among the 80 episodes whose original action sequences succeed

when re-executed in the simulator, 69 unmodified regenerations are judged successful; after the intervention, only 4 of these 69 (0.058) judgments change to failure. The mean pixel difference for this true-positive subset is similarly small at $0.56$. Among episodes whose unmodified regenerations are judged successful, using intervention response to distinguish true positives from false positives yields an AUROC of $0.50$. These results show that the world model is largely insensitive to the grasp-release intervention even when the corresponding action sequence changes from success to failure in the simulator. dWorldEval (Li et al., 2026) reports a related tendency for world-model evaluators to self-correct failures into successes; here we quantify this behavior using changed judgments, pixel difference, and discriminability.

### 3.4 Distance to the Demonstrations Across Candidates

The grasp-release intervention shows that the world model can depict outcomes inconsistent with the executed actions. Since the world model was fine-tuned on the task demonstrations, we hypothesize that its generated rollouts remain biased toward those demonstrations. Under this hypothesis, candidates whose generated rollouts stay closer to the demonstrations should tend to receive higher evaluator scores. We test this by computing, for each candidate, the $k$-nearest-neighbor distance from its generated observation-action pairs to the training demonstrations in a standardized embedding and taking the 90th percentile as $d$ (Appendix A.4).

Across all 31 candidates, $d$ is strongly associated with $\hat{p}$ (Pearson $r = -0.86$; Spearman $\rho = -0.87$; linear $R^2 = 0.73$; Figure 1(b)). The association is similar among the 24 ACT and Diffusion Policy candidates (Pearson $r = -0.86$; Spearman $\rho = -0.88$; linear $R^2 = 0.73$). After controlling linearly for $d$ within these 24 candidates, the partial correlation between $\hat{p}$ and $p$ is $+0.38$, with a 95% confidence interval of $[-0.04, +0.69]$. Thus, demonstration distance accounts for a large fraction of the variation in evaluator score, but these results do not establish that distance alone accounts for its association with success rate.

To compare proximity inside the world model with that of the candidate's behavior in the simulator, we compute the same distance from simulator rollouts, $d_{\mathrm{sim}}$. Across all 31 candidates, $d_{\mathrm{sim}}$ is strongly associated with $p$ (Pearson $r = -0.81$; Spearman $\rho = -0.94$; $R^2 = 0.66$) but only weakly with $\hat{p}$ (Pearson $r = -0.31$; Spearman $\rho = -0.30$; $R^2 = 0.10$ ). Among the 24 ACT and Diffusion Policy candidates, however, $d_{\mathrm{sim}}$ is also strongly associated with $\hat{p}$ ($R^2 = 0.62$). Thus the association between evaluator score and demonstration proximity is not specific to world-model rollouts.

### 3.5 Test-Time Perturbations Within One Candidate

The across-candidate association between evaluator score and demonstration proximity does not establish whether the same relationship holds within a fixed candidate. To test this, we apply test-time action perturbations while keeping the candidate parameters unchanged and measure the resulting changes in $p$, $\hat{p}$, and $d$. With its parameters frozen, we apply test-time action perturbations and measure the resulting changes in $p$, $\hat{p}$, and $d$. If the score tracks the action consequences, it should decrease when success decreases. If the score remains associated with demonstration proximity, a decrease in success need not lower the score when $d$ remains nearly unchanged.

We use one ACT checkpoint as the base and apply four test-time perturbations (Figure 1(c)). Delaying the grasp by eight steps keeps the arm trajectory and makes the grasp fail, while leaving $d$ nearly unchanged. The control perturbations, by contrast, change the trajectory itself. Scaling the actions by $0.6$ or $0.8$ changes the arm trajectory, while a two-step observation delay makes the policy act on stale observations.

For the trajectory-changing perturbations, decreases in $p$ are accompanied by decreases in $\hat{p}$, and increases in $d$: action scale $0.6$ gives $p = 0.49$ and $\hat{p} = 0.46$ with $d$ up by $0.18$, while observation delay 2 gives $p = 0.00$ and $\hat{p} = 0.02$, with $d$ up by $0.85$. Under the grasp delay, however, $p$ falls from $0.91$ to $0.37$ while $\hat{p}$ rises from $0.74$ to $0.84$. Generated videos under the grasp delay change little from those of the base, with a mean absolute pixel difference of $2.60$, compared with $7.57$ and $8.98$ for the two trajectory-changing perturbations. The generated videos remain visually similar to the base rollouts under the grasp delay (Figures 4 and 5). Across the five conditions, $\hat{p}$ decreases only when $d$ also increases, whereas the grasp delay lowers $p$ while leaving $d$ nearly unchanged.

Applying the same grasp delay to four additional ACT checkpoints reduces $p$ by 0.35–0.55, while $\hat{p}$ never decreases and the absolute change in $d$ is at most $0.022$ (Table 4) in Appendix A.5.

In summary, these results show that correlation with success does not imply sensitivity to action consequences. The evaluator score is strongly associated with demonstration proximity across candidates, yet it changes little under an intervention that changes the action consequence and can even increase when success decreases under the grasp-delay perturbation.

# 4 Identifiability Limits Without Real-World Trials

The empirical audit shows that correlation with success can coexist with insensitivity to action consequences. We ask whether this failure is specific to the evaluated world model or reflects a more general limitation under insufficient demonstration support. We formalize the latter for any evaluator that may use arbitrary prior information and candidate descriptions but performs no additional real-world trials. We show that insufficiently supported action differences cannot be reliably ranked and that perfect rank correlation can coexist with intervention insensitivity. All proofs are in Appendix B.

## 4.1 Setting

The impossibility result considers pairs of environments that differ only in the consequence of an action that is rarely taken in the demonstrations. We consider an episodic environment $M = (\mathcal{S}, \mathcal{A}, P_M, \rho_0, H, \mathsf{succ})$ with transition kernel $P_M$, horizon $H$, and success function succ. We write $p_M(\pi)$ for the success probability of a policy $\pi$ in $M$. The initial-state distribution $\rho_0$ is shared by the two environments. The demonstration policy $\beta$ does not observe the environment identity, so its action distribution conditioned on a history is the same in both environments. The demonstration set $D_n$ consists of $n$ independent episodes drawn from the episode distribution $P_M^\beta$, each with its terminal success label. We write $P_M^{\beta,n}$ for the distribution of $D_n$.

An evaluator is a measurable function of all information available to it,

$$\hat{i} = \mathcal{E}(B, D_n, \Pi, U), \tag{2}$$

returning the index of the candidate it judges better. Here, $B$ denotes fixed prior information, including the weights of a pretrained world model, the task description, and the success function. The candidate description $\Pi = (\pi_1, \pi_2)$ is complete, so the evaluator may inspect the candidates' weights and query their outputs on any history. The internal randomness $U$ is independent of the environment and the data. The evaluator may perform arbitrary computation using these inputs, but cannot obtain additional trials from the target environment. Because $B$ and $\Pi$ are identical across the environments being compared and $U$ is environment-independent, only $D_n$ carries environment-specific information. Therefore, world-model evaluation with a VLM judge is a special case of this class, with the pretrained world model included in $B$ and fine-tuning, rollouts, and judging in $\mathcal{E}$.

## 4.2 An Impossibility Result

The theorem considers a branch state that both candidates may reach and where they take different actions. The two environments differ only in the consequence of one of these actions, which the demonstration policy selects with probability $\varepsilon$ at the branch. We write $q_\beta$ and $q_\pi$ for the probabilities that $\beta$ and each candidate reach the branch, respectively. The success gap between the two actions has magnitude $\gamma$ and opposite signs in the two environments.

**Theorem 1** (Minimax impossibility of candidate ranking under partial action support)**.** *Let $\gamma \in (0, 1/4]$. For any $q_\beta, q_\pi \in (0, 1]$ and $\varepsilon \in [0, 1]$ there exist environments $M_0, M_1$, a demonstration policy $\beta$, and candidates $\pi_1, \pi_2$ such that*

1. $D_{\mathrm{KL}}\big(P_{M_0}^{\beta,n} \,\big\|\, P_{M_1}^{\beta,n}\big) \le 16\, n\, q_\beta\, \varepsilon\, \gamma^2$;
2. *the candidates' ranking reverses between the two environments:* $p_{M_0}(\pi_1) - p_{M_0}(\pi_2) \ge \gamma q_\pi$ *and* $p_{M_1}(\pi_2) - p_{M_1}(\pi_1) \ge \gamma q_\pi$;

*3. every evaluator of the form in Eq. (2) satisfies*

$$\max_{j\in\{0,1\}} \Pr_{M_j}\big[\hat{i} \neq i_j^*\big] \;\geq\; \tfrac{1}{2}\Big(1 - \sqrt{8\, n\, q_\beta\, \varepsilon\, \gamma^2}\Big)_+,$$

*where $i_j^*$ is the index of the better candidate in $M_j$.*

*In particular, if $\varepsilon = 0$, the worst-case error is at least $1/2$ for every $n$. Moreover, for $\delta < 1/4$, worst-case error at most $\delta$ requires $n\, q_\beta\, \varepsilon \geq \frac{1}{16\gamma^2} \log \frac{1}{4\delta}$.*

The proof uses a two-point Le Cam construction. Related lower bounds are known for offline reinforcement learning and policy selection (Foster et al., 2022; Liu et al., 2023b), but our access model additionally allows a pretrained world model, the success function, and the candidates' weights. Two candidates suffice because any ranking over a candidate set induces pairwise rankings.

Two consequences are particularly relevant. First, the bound depends on the demonstrations through $nq_\beta\varepsilon$, the expected number of episodes that both reach the branch and take the distinguishing action. We denote this effective support count by $n_{\text{supp}}$. Thus, identifiability depends on $n_{\text{supp}}$, not on $n$ alone. Demonstrations that do not contain this action do not improve the bound, regardless of their number. Second, the bound does not depend on fixed prior information $B$ or on how $\mathcal{E}$ uses the candidate descriptions $\Pi$. Therefore, a better world model or judge cannot remove the lack of information about an unsupported action; demonstrations containing that action or additional target environment trials are required. The bound already holds with a perfect judge. The grasp-release intervention exhibits the corresponding failure mode: the evaluator remains insensitive to an action consequence that is unsupported by the demonstrations.

### 4.3 Perfect Rank Correlation Without Sensitivity

A common validation criterion for world-model evaluators is rank agreement between estimated and measured success rates across a candidate's family. We show that even perfect rank agreement does not imply sensitivity to action consequences. Let $S(\pi)$ denote the probability that policy $\pi$ reaches the branch. An intervention $T$ is $S$-preserving for a family if $S(T\pi) = S(\pi)$ for every $\pi$ in the family.

**Proposition 1** (Rank correlation does not certify sensitivity)**.** *For any $K \geq 3$ and $\eta > 0$, there exist an environment, a candidate family $\{\pi^{(k)}\}_{k=1}^K$, a score $\hat{p}$ that depends only on $S$, and an $S$-preserving intervention $T$ such that (a) the Spearman rank correlation between $\hat{p}(\pi^{(k)})$ and $p(\pi^{(k)})$ over the family equals $1$, while (b)*

$$\hat{p}(T\pi^{(k)}) = \hat{p}(\pi^{(k)}) \;\text{ and }\; p(\pi^{(k)}) - p(T\pi^{(k)}) > 1 - \eta \tag{3}$$

*for every $k$.*

Rank correlation over a candidate family measures agreement only on the differences that vary within that family. Therefore, an evaluator can achieve perfect rank correlation while remaining completely insensitive to an intervention along an unvaried dimension.

## 5 ResGap: Coarse Screening Without World-Model Rollouts

Theorem 1 shows that candidate differences unsupported by the demonstrations cannot in general be resolved without additional target-environment trials. We therefore study ResGap (Residual Adaptation Gap) as an empirical heuristic for the narrower task of screening imitation-policy checkpoints before those trials. Rather than using the world model to approximate each candidate's deployment distribution, ResGap compares adaptation to observed and world-model-regenerated demonstrations. ResGap requires neither closed-loop candidate rollouts in the world model nor a VLM judge.

Let $D = \{(x_{j,t}, a_{j,t})\}_{j,t}$ denote the image–action pairs from the task demonstrations, where $j$ indexes demonstrations and $t$ timesteps, and $T_j$ is the length of demonstration $j$, and let $W$ denote the frozen world model used in the evaluator. For a subset of demonstrations, $W$ regenerates the image sequence from the original initial frame and action sequence, $\tilde{x}_{j,1:T_j} = W(x_{j,0}, a_{j,0:T_j-1})$,

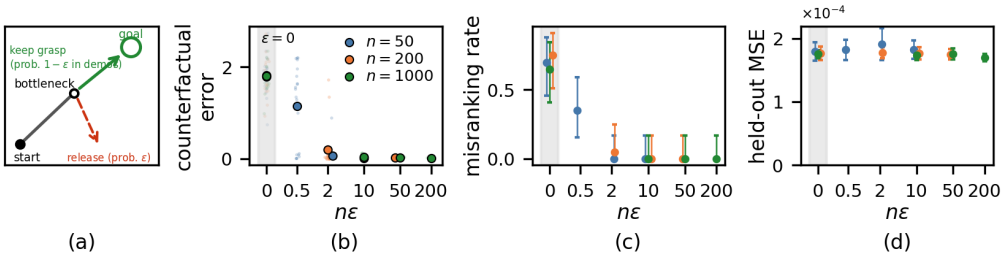


Figure 2: Controlled experiment against $n\varepsilon$, the expected number of demonstrations containing a release. (a) Transport task. (b) Counterfactual endpoint error for $n \in \{50, 200, 1000\}$, with means over 20 seeds (large markers), individual seeds (small points), and $\varepsilon = 0$ shaded. (c) Misranking rate with Clopper–Pearson 95% intervals. (d) Held-out MSE (mean $\pm 1$ standard deviation).

with $\tilde{x}_{j,0} = x_{j,0}$, yielding $\tilde{D} = \{(\tilde{x}_{j,t}, a_{j,t})\}_{j,t}$. ResGap compares adaptation magnitudes obtained from $D$ and $\tilde{D}$ under a fixed fine-tuning procedure.

Because real and regenerated videos can differ in appearance even when depicting the same motion, we insert a residual input adapter $A$ before the candidate, $\pi_{\theta,A}(x) = \pi_\theta(A(x))$, initialized to the identity; $\theta_0$ denotes the candidate's trained parameters. For each dataset $D' \in \{D, \tilde{D}\}$, we first optimize $A$ with $\theta = \theta_0$ fixed, then continue optimizing $A$ while fine-tuning $\theta$ from $\theta_0$. Let $u = \hat{\theta}_D - \theta_0$ and $v = \hat{\theta}_{\tilde{D}} - \theta_0$ denote the resulting candidate-parameter updates. We define ResGap as

$$\mathrm{ResGap}(\pi) = \frac{\|v\|_F - \|u\|_F}{\|\theta_0\|_F}. \tag{4}$$

ResGap is an empirical contrast in adaptation magnitude: larger values indicate greater parameter adaptation to regenerated than to real demonstrations under the specified procedure, without certifying performance or demonstration support.

**Use for coarse screening.** ResGap is intended for coarse screening rather than final policy selection. Given previously evaluated checkpoints, we fit a simple regression from ResGap to success rate and use the resulting predictions to prioritize new candidates for target-environment evaluation. The final comparison is left to target-environment trials. This use does not circumvent Theorem 1: ResGap cannot certify candidate differences whose consequences are unsupported by the available demonstrations.

## 6 Experiments

### 6.1 The Identifiability Limit in a Controlled Environment

To test the identifiability limit, we construct a two-dimensional transport task in which the demonstration policy releases the object at a bottleneck with probability $\varepsilon$ and otherwise keeps the grasp (Figure 2(a)). A release causes failure, so $\varepsilon$ directly controls support for the distinguishing action.

We evaluate four candidates: two that never release and two that release, the latter having zero success rate. For each $\varepsilon \in \{0, 0.01, 0.05, 0.2\}$ and $n \in \{50, 200, 1000\}$, we train a one-step dynamics model over 20 seeds and score the candidates from world-model rollouts using the known goal condition, excluding judge error (240 models; Appendix D). We measure counterfactual endpoint error after a forced release, candidate misranking, and held-out dynamics loss.

Counterfactual endpoint error decreases with $n\varepsilon$, but not with $n$ when $\varepsilon = 0$ (Figure 2(b)). In the unsupported regime, releasing candidates receive inflated scores and are misranked (Figure 2(c)); no misranking is observed for $n\varepsilon \geq 10$. Held-out dynamics loss remains similar across conditions (Figure 2(d)), so standard validation does not reveal the missing action consequence.

Table 2: Evaluation on 17 ACT checkpoints, leave-one-training-series-out. Brackets are bootstrap 95% intervals over checkpoints ($B = 2000$).

| Method | Rank correlation | MAE |
|---|---|---|
| World-model evaluator | $0.71$ $[0.26, 0.94]$ | $0.344$ $[0.254, 0.444]$ |
| Validation loss | $0.74$ $[0.19, 0.95]$ | $0.179$ $[0.088, 0.327]$ |
| Number of demonstrations | $0.47$ $[-0.62, 0.81]$ | $0.242$ $[0.137, 0.446]$ |
| Training epochs | $0.29$ $[-0.59, 0.68]$ | $0.295$ $[0.170, 0.447]$ |
| $\|u\|$ only (real data) | $0.80$ $[0.19, 0.94]$ | $0.169$ $[0.076, 0.324]$ |
| $\|v\|$ only (generated data) | $-0.39$ $[-0.86, 0.73]$ | $0.331$ $[0.140, 0.536]$ |
| ResGap | $\mathbf{0.95}$ $[0.72, 0.99]$ | $\mathbf{0.088}$ $[0.037, 0.204]$ |

### 6.2 ResGap for Coarse Checkpoint Screening

We evaluate ResGap on 17 ACT checkpoints from four training series, using simulator success rate as ground truth. Under leave-one-training-series-out evaluation, a linear regression from ResGap to success rate is fit on three series and applied to the fourth. We report Spearman rank correlation and mean absolute error (MAE) on the held-out predictions; implementation details are in Appendix C.2. We compare ResGap with the world-model evaluator, validation loss, number of demonstrations, and training epochs; we also report its two adaptation terms, $\|u\|$ and $\|v\|$, separately as ablations.

ResGap attains a rank correlation of $0.95$ and an MAE of $0.088$, compared with $0.74$ and $0.179$ for validation loss and $0.71$ and $0.344$ for the world-model evaluator (Table 2). The individual terms perform worse in point estimates. Bootstrap intervals overlap for rank correlation across all methods; for MAE, ResGap overlaps validation loss but not the world-model evaluator.

We additionally evaluate four checkpoints from two training series not included in the 17-checkpoint analysis. A regression fit on all 17 checkpoints gives an MAE of $0.130$ on these four checkpoints; three predictions are within $0.05$ of the measured success rate, while one checkpoint with $p = 0.04$ is predicted at $0.47$. This failure illustrates why ResGap is used only for coarse screening rather than to certify candidate performance. Matched-pool variants give lower leave-one-series-out rank correlations and higher MAE; Appendix C.2 reports these analyses.

## 7 Discussion

Correlation with measured success rate shows agreement on the evaluated candidates, but not sensitivity to action consequences. Such sensitivity depends on support for the action that separates the candidates rather than on the total number of demonstrations. Model-level remedies and additional failure demonstrations do not remove this dependence when that action remains unsupported.

Future world models may generalize far beyond their task-specific demonstrations. As foundation models scale, they may acquire sufficiently broad physical and behavioral priors to predict consequences of actions unsupported in the target demonstrations. Theorem 1 is a worst-case identifiability result and does not rule out such generalization; rather, it separates predictive capability from identifiability. Without target-environment evidence for the distinguishing action, the demonstrations alone cannot certify that the prior is correct for that environment. We evaluate ResGap as an empirical coarse-screening heuristic from demonstrations without rollouts or a judge, but it does not certify performance outside demonstrated support. Unresolved distinctions are therefore left to real-world trials.

The main audit is limited to LIBERO-Object, and ResGap is evaluated only on checkpoints from one policy family. Its rank-correlation interval overlaps those of the baselines, and a near-zero-success checkpoint is substantially overpredicted. Overall, our results delimit what evaluation without additional real-world trials can establish: agreement on demonstrated behavior does not certify sensitivity to unsupported action consequences.

### AI USE STATEMENT

We used generative AI tools to assist with hypothesis refinement, theoretical development and proof construction, methodology and experimental design, implementation and experiment execution, data processing and analysis, and interpretation of results. We also used generative AI tools for literature search and citation checking, manuscript drafting, translation and editing, and figure preparation. The authors reviewed the AI-assisted contributions included in the final manuscript and take responsibility for the final content, including the claims, proofs, experimental results, code, and presentation.

### ETHICS STATEMENT

This work does not involve human participants, personal data, or real-world deployment. All experiments are conducted in simulation using robot-manipulation benchmarks and pretrained models.

### REPRODUCIBILITY STATEMENT

The evaluator configuration and audit protocol are described in Appendix A, and complete proofs of the theoretical results are provided in Appendix B. Appendix C specifies the ResGap implementation and checkpoint-screening protocol, and Appendix D gives the full setup of the controlled experiment. Additional evaluator variants are described in Appendix E.

## REFERENCES

Pranav Atreya, Karl Pertsch, Tony Lee, Moo Jin Kim, Arhan Jain, Artur Kuramshin, Cyrus Neary, Edward S. Hu, Kanav Arora, Kirsty Ellis, Luca Macesanu, Matthew Leonard, Meedeum Cho, Ozgur Aslan, Shivin Dass, Tony Wang, Xingfang Yuan, Abhishek Gupta, Dinesh Jayaraman, Glen Berseth, Kostas Daniilidis, Roberto Martín-Martín, Youngwoon Lee, Percy Liang, Chelsea Finn, and Sergey Levine. RoboArena: Distributed real-world evaluation of generalist robot policies. In Joseph Lim, Shuran Song, and Hae-Won Park (eds.), *Proceedings of The 9th Conference on Robot Learning*, volume 305 of *Proceedings of Machine Learning Research*, pp. 336–364. PMLR, 27–30 Sep 2025. URL https://proceedings.mlr.press/v305/atreya25a.html.

Apurva Badithela, David Snyder, Lihan Zha, Joseph Mikhail, Matthew O'Kelly, Anushri Dixit, and Anirudha Majumdar. Reliable and scalable robot policy evaluation with imperfect simulators. *arXiv preprint arXiv:2510.04354*, 2025.

Christina Baek, Yiding Jiang, Aditi Raghunathan, and Zico Kolter. Agreement-on-the-line: predicting the performance of neural networks under distribution shift. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA, 2022. Curran Associates Inc. ISBN 9781713871088.

Jinglin Chen and Nan Jiang. Information-theoretic considerations in batch reinforcement learning. In *International conference on machine learning*, pp. 1042–1051. PMLR, 2019.

Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, 44(10-11):1684–1704, 2025.

Weijian Deng and Liang Zheng. AutoEval: Are labels always necessary for classifier accuracy evaluation? *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(3):1868–1880, 2021.

Dylan J Foster, Akshay Krishnamurthy, David Simchi-Levi, and Yunzong Xu. Offline reinforcement learning: Fundamental barriers for value function approximation. In Po-Ling Loh and Maxim Raginsky (eds.), *Proceedings of Thirty Fifth Conference on Learning Theory*, volume 178 of *Proceedings of Machine Learning Research*, pp. 3489–3489. PMLR, 02–05 Jul 2022. URL https://proceedings.mlr.press/v178/foster22a.html.

Shenyuan Gao, William Liang, Kaiyuan Zheng, Ayaan Malik, Seonghyeon Ye, Sihyun Yu, Wei-Cheng Tseng, Yuzhu Dong, Kaichun Mo, Chen-Hsuan Lin, et al. Dreamdojo: A generalist robot world model from large-scale human videos. *arXiv preprint arXiv:2602.06949*, 2026.

Saurabh Garg, Sivaraman Balakrishnan, Zachary C Lipton, Behnam Neyshabur, and Hanie Sedghi. Leveraging unlabeled data to predict out-of-distribution performance. *arXiv preprint arXiv:2201.04234*, 2022.

Yanjiang Guo, Lucy Shi, Jianyu Chen, and Chelsea Finn. Ctrl-world: A controllable generative world model for robot manipulation. In *International Conference on Learning Representations*, volume 2026, pp. 6121–6138, 2026.

Siqiao Huang, Jialong Wu, Qixing Zhou, Shangchen Miao, and Mingsheng Long. Vid2world: Crafting video diffusion models to interactive world models. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 10763–10806, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/1214d67bb7fa38d40e0b3c2be677e39b-Paper-Conference.pdf.

Byeongguk Jeon, Seonghyeon Ye, JaeHyeok Doo, Sungdong Kim, Minjoon Seo, Hyungmok Son, and Kimin Lee. RoboWorld: Fast and reliable neural simulators for generalist robot policy evaluation. *arXiv preprint arXiv:2607.01060*, 2026.

Nan Jiang and Lihong Li. Doubly robust off-policy value evaluation for reinforcement learning. In Maria Florina Balcan and Kilian Q. Weinberger (eds.), *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pp. 652–661, New York, New York, USA, 20–22 Jun 2016. PMLR. URL https://proceedings.mlr.press/v48/jiang16.html.

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 Pulkit Agrawal, Oliver Kroemer, and Wolfram Burgard (eds.), *Proceedings of The 8th Conference on Robot Learning*, volume 270 of *Proceedings of Machine Learning Research*, pp. 2679–2713. PMLR, 06–09 Nov 2025. URL https://proceedings.mlr.press/v270/kim25c.html.

Xuanlin Li, Kyle Hsu, Jiayuan Gu, Oier Mees, Karl Pertsch, Homer Rich Walke, Chuyuan Fu, Ishikaa Lunawat, Isabel Sieh, Sean Kirmani, Sergey Levine, Jiajun Wu, Chelsea Finn, Hao Su, Quan Vuong, and Ted Xiao. Evaluating real-world robot manipulation policies in simulation. In Pulkit Agrawal, Oliver Kroemer, and Wolfram Burgard (eds.), *Proceedings of The 8th Conference on Robot Learning*, volume 270 of *Proceedings of Machine Learning Research*, pp. 3705–3728. PMLR, 06–09 Nov 2025a. URL https://proceedings.mlr.press/v270/li25c.html.

Yaxuan Li, Yichen Zhu, Junjie Wen, Chaomin Shen, and Yi Xu. Worldeval: World model as real-world robot policies evaluator. *arXiv preprint arXiv:2505.19017*, 2025b.

Yaxuan Li, Zhongyi Zhou, Yefei Chen, Yaokai Xue, and Yichen Zhu. dworldeval: Scalable robotic policy evaluation via discrete diffusion world model. *arXiv preprint arXiv:2604.22152*, 2026.

Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. Libero: Benchmarking knowledge transfer for lifelong robot learning. *Advances in Neural Information Processing Systems*, 36:44776–44791, 2023a.

Vincent Liu, Prabhat Nagarajan, Andrew Patterson, and Martha White. When is offline policy selection sample efficient for reinforcement learning? *arXiv preprint arXiv:2312.02355*, 2023b.

Ajay Mandlekar, Danfei Xu, Josiah Wong, Soroush Nasiriany, Chen Wang, Rohun Kulkarni, Li Fei-Fei, Silvio Savarese, Yuke Zhu, and Roberto Martín-Martín. What matters in learning from offline human demonstrations for robot manipulation. In Aleksandra Faust, David Hsu, and Gerhard Neumann (eds.), *Proceedings of the 5th Conference on Robot Learning*, volume 164 of *Proceedings of Machine Learning Research*, pp. 1678–1690. PMLR, 08–11 Nov 2022. URL https://proceedings.mlr.press/v164/mandlekar22a.html.

John P Miller, Rohan Taori, Aditi Raghunathan, Shiori Sagawa, Pang Wei Koh, Vaishaal Shankar, Percy Liang, Yair Carmon, and Ludwig Schmidt. Accuracy on the line: on the strong correlation between out-of-distribution and in-distribution generalization. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 7721–7735. PMLR, 18–24 Jul 2021.

Julian Quevedo, Ansh Kumar Sharma, Yixiang Sun, Varad Suryavanshi, Percy Liang, and Sherry Yang. WorldGym: World model as an environment for policy evaluation. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 78932–78957, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/7f5e909ac0324db03506b380c695ffaf-Paper-Conference.pdf.

David Snyder, Asher James Hancock, Apurva Badithela, Emma Dixon, Patrick Miller, Rares Andrei Ambrus, Anirudha Majumdar, Masha Itkina, and Haruki Nishimura. Is your imitation learning policy better than mine? policy comparison with near-optimal stopping. *arXiv preprint arXiv:2503.10966*, 2025.

Shengpu Tang and Jenna Wiens. Model selection for offline reinforcement learning: Practical considerations for healthcare settings. In *Machine Learning for Healthcare Conference*, pp. 2–35. PMLR, 2021.

Gemini Robotics Team, Krzysztof Choromanski, Coline Devin, Yilun Du, Debidatta Dwibedi, Ruiqi Gao, Abhishek Jindal, Thomas Kipf, Sean Kirmani, Isabel Leal, et al. Evaluating gemini robotics policies in a veo world simulator. *arXiv preprint arXiv:2512.10675*, 2025.

Qwen Team. Qwen2.5-VL technical report. *arXiv preprint arXiv:2502.13923*, 2025.

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

Wei-Cheng Tseng, Gashon Hussein, Yuzhu Dong, Allen Z Ren, Lucy X Shi, XuDong Wang, Sergey Levine, Zhaoshuo Li, Jinwei Gu, Florian Shkurti, et al. SC3-Eval: Evaluating robot foundation models via self-consistent video generation. *arXiv preprint arXiv:2606.18610*, 2026.

Alexandre B. Tsybakov. *Introduction to Nonparametric Estimation*. Springer, 2009.

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, et al. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

Ruosong Wang, Dean P Foster, and Sham M Kakade. What are the statistical limits of offline rl with linear function approximation? *arXiv preprint arXiv:2010.11895*, 2020.

Yixuan Wang, Rhythm Syed, Fangyu Wu, Mengchao Zhang, Aykut Onol, Jose Barreiros, Hooshang Nayyeri, Tony Dear, Huan Zhang, and Yunzhu Li. Interactive world simulator for robot policy training and evaluation. *arXiv preprint arXiv:2603.08546*, 2026.

Yaodong Yu, Zitong Yang, Alexander Wei, Yi Ma, and Jacob Steinhardt. Predicting out-of-distribution error with the projection norm. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 25721–25746. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/yu22i.html.

Andrea Zanette. Exponential lower bounds for batch reinforcement learning: Batch rl can be exponentially harder than online rl. In *International Conference on Machine Learning*, pp. 12287–12297. PMLR, 2021.

Siyuan Zhang and Nan Jiang. Towards hyperparameter-free policy selection for offline reinforcement learning. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 12864–12875. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/6add07cf50424b14fdf649da87843d01-Paper.pdf.

Tony Z Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning fine-grained bimanual manipulation with low-cost hardware. *arXiv preprint arXiv:2304.13705*, 2023.

Zhiyuan Zhou, Pranav Atreya, You Liang Tan, Karl Pertsch, and Sergey Levine. AutoEval: Autonomous evaluation of generalist robot manipulation policies in the real world. In Joseph Lim, Shuran Song, and Hae-Won Park (eds.), *Proceedings of The 9th Conference on Robot Learning*, volume 305 of *Proceedings of Machine Learning Research*, pp. 1997–2017. PMLR, 27–30 Sep 2025. URL https://proceedings.mlr.press/v305/zhou25a.html.

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

# A Details of the Evaluator Audit

## A.1 Evaluator, Candidates, and Protocol

**World model.** The world model is Wan2.1-1.3B (Wan et al., 2025) fine-tuned on the 450 training demonstrations of LIBERO-Object (Liu et al., 2023a) to generate the next frames given actions. The action conditioning enters as a two-channel control map concatenated to the video latent along channels; the patch embedding is extended from 16 to 18 input channels with the added channels initialized to zero; the extended patch embedding is fully trainable. The DiT blocks are fine-tuned with LoRA of rank 32 and $\alpha = 32$ on the query, key, value, and output projections of both attention layers and on both layers of the feed-forward network; the VAE and the text encoder are frozen.

Fine-tuning proceeds in three stages on clips of the demonstrations. The first stage runs 8000 steps at learning rate $10^{-4}$ on clips of 17 frames without augmentation. The second stage runs 3000 steps at learning rate $5 \times 10^{-5}$ on clips of 5 or 17 frames with context-degradation augmentation (ctx_aug $= 0.5$). The third stage runs 8000 steps at learning rate $5 \times 10^{-5}$ with the same clips and augmentation, plus event-focused sampling that prioritizes clips around grasps and releases (event_bias $= 0.5$). All stages use batch size 4 with gradient accumulation of 2 and drop the text condition with probability 0.1. The checkpoint after the third stage is used throughout the paper.

**Judge.** The judge is Qwen2.5-VL-7B (Team, 2025). Each generated episode is subsampled to 8 frames at equal intervals, and the episode is labeled by majority vote over the following three prompts, where {task} is the task instruction.

1. *The robot was instructed to: "{task}". Based on the video frames, did the robot successfully complete this task? Placing the object into the basket, or lowering it into the basket in the last frames, counts as success. Answer with only "yes" or "no".*
2. *Task given to the robot: "{task}". Looking at these frames from start to end, did the robot pick up the correct object and get it into (or directly above, about to drop into) the basket by the end? Reply "yes" or "no" only.*
3. *These are frames of a robot manipulation episode. Instruction: "{task}". Judge whether by the final frames the instructed object has been transported to the basket. Answer only "yes" or "no".*

**Episodes.** Each rollout starts from the initial frame of a simulator episode. The ground truth $p$ is the success rate of the same candidate over simulator episodes with the same initial states.

**Candidates.** The 31 candidates comprise 17 ACT checkpoints from four training series, 7 Diffusion Policy checkpoints, 3 BC-RNN, 3 BC-Transformer, and 1 OpenVLA. ACT and Diffusion Policy output action chunks and match the action interface of the world model.

## A.2 Error decomposition

For each candidate, we record the action sequence produced inside the world model in every episode and re-execute it in the simulator from the same initial state. $p_{\mathrm{cf}}$ is the success rate of the re-executed

Table 3: Distance analysis using simulator rollouts. Entries are Pearson $r$ / Spearman $\rho$ / linear $R^2$ unless otherwise noted. Partial correlations are reported with 95% confidence intervals (Fisher $z$).

| | 31 candidates | 24 ACT+DP candidates |
|---|---|---|
| $d_{\text{sim}}$ vs. $p$ | $-0.813/-0.937/0.661$ | $-0.803/-0.935/0.645$ |
| $d_{\text{sim}}$ vs. $\hat{p}$ | $-0.309/-0.297/0.096$ | $-0.787/-0.753/0.620$ |
| $d_{\text{sim}}$ vs. $d$ | $+0.618/+0.573/0.382$ | $+0.926/+0.910/0.857$ |
| Partial corr. $(\hat{p}, p \mid d_{\text{sim}})$ | — | $+0.251$ $[-0.180, +0.601]$ |

episodes, and $j_{\text{cf}}$ is the fraction of re-executed episodes that the same judge labels as successful. The entries of Table 1 are the three terms of Eq. (1) averaged over the candidates of each family.

### A.3 Intervention on the action sequence

We take episodes from the 24 ACT and Diffusion Policy candidates that the judge labeled as successful. In each action sequence, we insert a grasp release while the object is being transported and keep the rest of the sequence unchanged. The modified sequence is re-executed in the simulator from the same initial state, and the video is regenerated from the modified sequence with the same initial frame. For the success-to-failure analysis, the changed-judgment rate is the fraction of episodes whose original action sequence succeeds in the simulator and whose unmodified regeneration is judged successful, for which the modified regeneration is judged unsuccessful. The pixel difference is the mean absolute difference per pixel between the original and the regenerated video, averaged over frames and episodes. Among episodes whose unmodified regenerations are judged successful, we use whether the judgment changes from success to failure under the intervention to distinguish true positives from false positives and report the resulting AUROC.

### A.4 Distance to the demonstrations

**Definition.** For each candidate, we collect observation–action pairs from its rollouts, embed each observation with a ResNet-18 encoder, and standardize both the observation embedding and the action using statistics from the reference demonstrations. For each pair we compute the mean distance to its $k = 10$ nearest neighbors among the pairs from 500 reference demonstrations, and define $d$ as the 90th percentile of these distances over the candidate's pairs. The world-model distance $d$ is computed from generated rollouts (140 episodes per candidate, including regenerated episodes).

**Simulator-side distance.** $d_{\text{sim}}$ uses the same definition and parameters on closed-loop rollouts in the simulator (tasks 00–09 $\times$ seeds 00–09, 100 episodes per candidate), which are also used to measure $p$. Table 3 reports the corresponding correlations using $d_{\text{sim}}$ in place of $d$, and Figure 3 shows $\hat{p}$ against $d_{\text{sim}}$. Across all 31 candidates, $d_{\text{sim}}$ is strongly associated with $p$ ($R^2 = 0.66$) but only weakly with $\hat{p}$ ($R^2 = 0.10$). Among the 24 ACT and Diffusion Policy candidates, $d_{\text{sim}}$ is strongly associated with both $p$ ($R^2 = 0.65$) and $\hat{p}$ ($R^2 = 0.62$), and closely tracks $d$ ($R^2 = 0.86$). The partial correlation between $\hat{p}$ and $p$ controlling for $d_{\text{sim}}$ is $0.25$ with a 95% confidence interval of $[-0.18, 0.60]$. Thus, the association between evaluator score and demonstration proximity is not specific to world-model rollouts.

### A.5 Test-time perturbations

The base policy is the ACT checkpoint trained on 45 demonstrations with seed 2 and selected at epoch 60. The action-scale perturbation multiplies each action by 0.6 or 0.8. Observation delay feeds the policy the observation from two steps earlier. Grasp delay postpones the gripper-closing command by eight steps while leaving the arm actions unchanged. Each perturbed policy is evaluated in the simulator and inside the world model with the same protocol as the base.

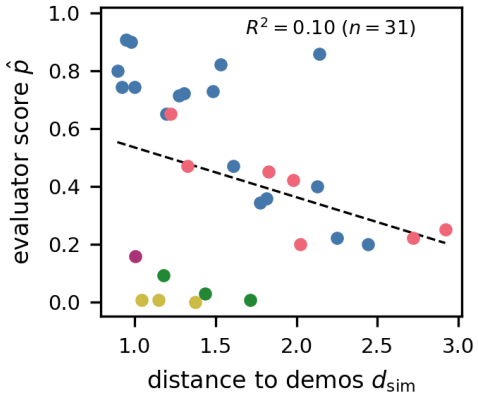


Figure 3: Evaluator score $\hat{p}$ against the simulator-side distance $d_{\text{sim}}$ for 31 candidates. Colors follow Figure 1. The dashed line is the linear regression ($R^2 = 0.10$).

Table 4: Grasp delay applied to four other ACT checkpoints. For each checkpoint, we report simulator success rate $p$, evaluator score $\hat{p}$, and demonstration distance $d$, with and without the eight-step grasp delay.

| Checkpoint | $p$ | $p_{\text{delay}}$ | $\hat{p}$ | $\hat{p}_{\text{delay}}$ | $d$ | $d_{\text{delay}}$ |
|---|---|---|---|---|---|---|
| ACT-45-e60 (run 1) | 0.60 | 0.25 | 0.74 | 0.81 | 1.18 | 1.20 |
| ACT-45-s0-e60 | 0.69 | 0.26 | 0.90 | 0.90 | 1.14 | 1.16 |
| ACT-45-s1-e60 | 0.83 | 0.28 | 0.91 | 0.91 | 1.17 | 1.18 |
| ACT-45-s2-e120 | 0.85 | 0.41 | 0.80 | 0.80 | 1.15 | 1.15 |

# B Theory: Statements and Proofs

## B.1 Setting

We consider an episodic environment $M = (\mathcal{S}, \mathcal{A}, P_M, \rho_0, H, \mathsf{succ})$. The initial state distribution $\rho_0$ is common to all environments compared. We write $p_M(\pi)$ for the success probability of a policy $\pi$. The demonstration policy $\beta$ does not observe the environment label and has the same action distribution given the same history. Trajectory observations include the terminal success label. The demonstration set $D_n$ consists of $n$ independent samples from the episode distribution $P_M^\beta$ of $\beta$.

An evaluator is a measurable map

$$\hat{\imath} = \mathcal{E}(B, D_n, \Pi, U), \tag{5}$$

where $B$ is fixed prior information (the weights of a pretrained world model, the task description, the success function, and the algorithm), $\Pi = (\pi_1, \pi_2)$ is the complete description of the candidate policies, so that the evaluator can read the candidates' weights and query their outputs for any history, and $U$ is internal randomness independent of the environment and the data. The evaluator performs no additional trials in the real environment. In the constructions below, $B$ and $\Pi$ are fixed deterministic objects common to both environments compared.

## B.2 Theorem 1 (Minimax impossibility of candidate ranking under partial action support)

**Theorem 1** (restated). *Let $\gamma \in (0, 1/4]$. For any $q_\beta, q_\pi \in (0, 1]$ and $\varepsilon \in [0, 1]$ there exist environments $M_0, M_1$, a demonstration policy $\beta$, and candidates $\pi_1, \pi_2$ such that the following hold.*

1. $D_{\mathrm{KL}}\big(P_{M_0}^{\beta,n} \,\big\|\, P_{M_1}^{\beta,n}\big) \le 16\, n\, q_\beta\, \varepsilon\, \gamma^2$.
2. *The true ranking is reversed:* $p_{M_0}(\pi_1) - p_{M_0}(\pi_2) \ge \gamma q_\pi$ *and* $p_{M_1}(\pi_2) - p_{M_1}(\pi_1) \ge \gamma q_\pi$.

*3. For every evaluator,* $\max_{j\in\{0,1\}} \Pr_{M_j}[\hat{i} \neq i_j^*] \;\geq\; \frac{1}{2}\Big(1 - \sqrt{8\, n\, q_\beta\, \varepsilon\, \gamma^2}\Big)_+$.

*In particular, if* $\varepsilon = 0$ *the worst-case error is at least* $1/2$ *for every* $n$*. Moreover, for* $\delta < 1/4$*, worst-case error at most* $\delta$ *requires* $n\, q_\beta\, \varepsilon \geq \frac{1}{16\gamma^2} \log \frac{1}{4\delta}$*. Writing* $\Delta = \gamma q_\pi$ *for the performance gap, this necessary condition reads* $n \gtrsim \frac{q_\pi^2}{q_\beta \varepsilon \Delta^2} \log \frac{1}{\delta}$.

*Proof.* **Construction.** Let the state space be $\{s_{\text{start}}, s_{\text{mid}}, s_{\text{null}}, s_{\text{succ}}, s_{\text{fail}}\}$ and $H = 2$. The terminal states are $s_{\text{null}}$, $s_{\text{succ}}$, and $s_{\text{fail}}$, with $\text{succ}(s_{\text{succ}}) = 1$ and $\text{succ}(s_{\text{null}}) = \text{succ}(s_{\text{fail}}) = 0$. At $s_{\text{start}}$, the action $a_{\text{go}}$ moves deterministically to $s_{\text{mid}}$, and every other action moves deterministically to the absorbing failure state $s_{\text{null}}$. At $s_{\text{mid}}$, the action $a_0$ moves to $s_{\text{succ}}$ and $s_{\text{fail}}$ with probability $1/2$ each in both environments, whereas $a_1$ moves to $s_{\text{succ}}$ with probability $\frac{1}{2} - \gamma$ in $M_0$ and $\frac{1}{2} + \gamma$ in $M_1$ (and to $s_{\text{fail}}$ otherwise). The two environments differ only in the transition at $(s_{\text{mid}}, a_1)$. The observation distributions after $s_{\text{null}}$ are common to both environments.

The demonstration policy $\beta$ takes $a_{\text{go}}$ at $s_{\text{start}}$ with probability $q_\beta$ and, conditional on reaching $s_{\text{mid}}$, takes $a_1$ with probability $\varepsilon$ and $a_0$ otherwise. Both candidates take $a_{\text{go}}$ at $s_{\text{start}}$ with probability $q_\pi$. Conditional on reaching $s_{\text{mid}}$, $\pi_1$ takes $a_0$ and $\pi_2$ takes $a_1$.

**Claim 2.** This follows from $p_{M_j}(\pi_1) = q_\pi/2$ (in both environments), $p_{M_0}(\pi_2) = q_\pi(\frac{1}{2} - \gamma)$, and $p_{M_1}(\pi_2) = q_\pi(\frac{1}{2} + \gamma)$.

**Claim 1.** The KL divergence of one episode is given by the chain rule over histories, $\text{KL} = \mathbb{E}_{P^\beta_{M_0}}\big[\sum_t \text{KL}(P_{M_0}(\cdot \mid H_t, A_t) \,\|\, P_{M_1}(\cdot \mid H_t, A_t))\big]$. Since the conditional action distributions of $\beta$ are identical in both environments, the action-selection terms contribute zero. The transition kernels differ only at $(s_{\text{mid}}, a_1)$, which is reached with probability $q_\beta\varepsilon$ in both environments. Hence

$$\text{KL}(P^\beta_{M_0} \| P^\beta_{M_1}) = q_\beta\varepsilon \cdot \text{KL}\big(\text{Ber}(\tfrac{1}{2} - \gamma) \,\|\, \text{Ber}(\tfrac{1}{2} + \gamma)\big) = q_\beta\varepsilon \cdot 4\gamma\, \text{artanh}(2\gamma). \tag{6}$$

Since $\text{artanh}(y) \leq 2y$ for $y \leq 1/2$, the right-hand side is at most $16 q_\beta \varepsilon \gamma^2$ for $\gamma \leq 1/4$. For $n$ independent samples, tensorization multiplies the bound by $n$.

**Claim 3.** Because $B$ and $\Pi$ are deterministic and common to both environments and $U$ is independent of $(M, D_n)$, the distribution of the evaluator's input depends on the environment only through $D_n$. Viewing the selection as a randomized test on the data, the data-processing inequality and Le Cam's two-point lemma give a worst-case error of at least $\frac{1}{2}(1 - \text{TV}(P^{\beta,n}_{M_0}, P^{\beta,n}_{M_1}))$, and Pinsker's inequality gives $\text{TV} \leq \sqrt{\text{KL}_n/2} \leq \sqrt{8 n q_\beta \varepsilon \gamma^2}$. If $\varepsilon = 0$, the two distributions coincide and $\text{TV} = 0$. The necessary condition follows from the Bretagnolle–Huber inequality $\text{TV} \leq 1 - \frac{1}{2} e^{-\text{KL}_n}$, which gives a worst-case error of at least $\frac{1}{4} e^{-16 n q_\beta \varepsilon \gamma^2}$. □

### B.3 Proposition 1 (Correlation on a calibration family does not guarantee counterfactual ranking)

A policy intervention $T$ is *$S$-preserving* for a candidate family if $S(T\pi) = S(\pi)$ for every member $\pi$ of the family. In the construction below, $S(\pi) = \Pr_\pi(\text{reach } s_{\text{mid}})$.

**Proposition 1** (restated)**.** *For any* $K \geq 3$ *and* $\eta > 0$ *there exist a calibration family* $\{\pi^{(k)}\}_{k=1}^K$*, an environment, an evaluator that reads only* $S$*, and an* $S$*-preserving intervention* $T$ *such that (a) the Spearman rank correlation between* $\hat{p}_k$ *and* $p_k$ *over the calibration family is* $1$*, and (b) the intervention leaves* $\hat{p}$ *unchanged while* $p(\pi^{(k)}) - p(T\pi^{(k)}) > 1 - \eta$ *for every* $k$*.*

*Proof.* Modify the construction of Theorem 1 so that, at $s_{\text{mid}}$, the action $a_0$ moves deterministically to $s_{\text{succ}}$ in both environments. Let the calibration family $\pi^{(k)}$ take $a_{\text{go}}$ at $s_{\text{start}}$ with probability $\sigma_k$ and $a_0$ at $s_{\text{mid}}$, with $1 - \eta < \sigma_1 < \cdots < \sigma_K < 1$. Then $p_k = \sigma_k$ is strictly monotone in $S_k = \sigma_k$, and any evaluator $\hat{p}_k = g(S_k)$ with $g$ strictly increasing on $\{S_k\}$ attains Spearman correlation $1$ over the family (there are no ties). Let the intervention $T$ replace only the conditional action after reaching $s_{\text{mid}}$ by $a_1$. The probability of reaching $s_{\text{mid}}$ is unchanged, so $T$ is $S$-preserving and $\hat{p}$ is unchanged. Letting $a_1$ move deterministically to $s_{\text{fail}}$ gives $p(T\pi^{(k)}) = 0$, so the drop equals $\sigma_k > 1 - \eta$. □

### B.4 A LINEARIZED CHARACTERIZATION OF THE BODY UPDATE

We characterize the two adaptation stages used in ResGap under a linearized policy body. For the first-stage adapter fitting, let $r_A(x) = f(\theta_0, A; x) - y$ and $J_A(x) = \partial f(\theta_0, A; x)/\partial A$. After fitting the adapter $\hat{A}$, define $z_i = \phi(\hat{A}(x_i))$ and $e_i = y_i - f_{\Theta_0}(\hat{A}(x_i))$, and collect them as

$$Z = [z_1, \dots, z_n] \in \mathbb{R}^{d\times n}, \qquad E = [e_1, \dots, e_n] \in \mathbb{R}^{m\times n}.$$

For the linear body $f_\Theta(z) = \Theta z$, write $\Delta = \Theta - \Theta_0 \in \mathbb{R}^{m\times d}$.

**Proposition 2** (Adapter orthogonality and body-update characterization)**.**

1. ***First-stage orthogonality.*** *If $L_n(A) = \frac{1}{2n}\sum_i \|r_A(x_i)\|_2^2$ is differentiable, $\hat{A}$ is an interior stationary point, and there is no regularization, then*
$$\frac{1}{n}\sum_i J_{\hat{A}}(x_i)^\top r_{\hat{A}}(x_i) = 0.$$

2. ***Second-stage minimum-norm solution.*** *Consider*
$$L(\Delta) = \frac{1}{2}\|\Delta Z - E\|_F^2, \qquad \dot{\Delta} = -(\Delta Z - E)Z^\top, \qquad \Delta(0) = 0.$$
*Let $Z = U\Sigma V^\top$ be its compact SVD. Then*
$$\Delta(t) = EV\Sigma^{-1}\big(I - e^{-\Sigma^2 t}\big)U^\top \longrightarrow \Delta_\infty = EZ^\dagger.$$
*Moreover, $\Delta_\infty$ is the unique minimum-Frobenius-norm minimizer of $\|\Delta Z - E\|_F^2$, and*
$$\|\Delta_\infty\|_F^2 = \operatorname{tr}\left(EZ^\dagger Z^{\dagger\top} E^\top\right).$$

3. ***Update identity.*** *For $g = v - u$,*
$$\|v\|_F^2 - \|u\|_F^2 = \|g\|_F^2 + 2\langle u, g\rangle_F.$$

*Proof.* (1) is the first-order condition itself. (2) follows by direct verification of the closed form: the time derivative of $\Delta(t)$ equals the right-hand side of the gradient flow, and each row of $\Delta(t)$ (transposed) lies in $\mathrm{col}(Z) \subseteq \mathbb{R}^d$ at all times. Any least-squares solution can be written as $\Delta = EZ^\dagger + K(I - ZZ^\dagger)$, and each row of the second term lies in $\mathrm{col}(Z)^\perp$; imposing that every row lies in $\mathrm{col}(Z)$ forces $K(I - ZZ^\dagger) = 0$, so $\Delta = EZ^\dagger$ is unique. (3) follows by expansion. □

The characterization in part 2 assumes that the fitted adapter $\hat{A}$ is held fixed during the second-stage analysis, so that $Z$ is fixed. It further assumes Euclidean gradient flow without momentum, Adam-type preconditioning, weight decay, or additional regularization, and a body that is linear in $\Theta$. These assumptions do not hold exactly in our implementation, where the adapter and nonlinear policy body are jointly optimized with Adam.

### B.5 IDENTIFICATION WITH JUDGE ERROR

We model judge errors by stratum-wise sensitivity and specificity: $\hat{p} = s_e p + (1 - s_p)(1 - p)$ with $(s_e, s_p) \in C$ and $J = s_e + s_p - 1$. Define the identified set

$$\mathcal{P}(\hat{p}, C) = \left\{\tfrac{\hat{p}+s_p-1}{J} \in [0,1] : (s_e, s_p) \in C,\ J \neq 0\right\} \cup \big([0,1] \text{ if } \exists (s_e, s_p) \in C :\ J = 0,\ s_e = \hat{p}\big). \tag{7}$$

**Proposition 3** (Partial identification through the judge channel)**.** *Under the notation above, the following statements hold.*

*(i) If $C = \{(s_e, s_p)\}$ is a singleton, then: if $J \neq 0$,*
$$p = \frac{\hat{p} + s_p - 1}{J}$$
*is point-identified; if $J = 0$ and $s_e = \hat{p}$, then $\mathcal{P} = [0,1]$ (no information); and if $J = 0$ and $s_e \neq \hat{p}$, then $\mathcal{P} = \varnothing$ (model misspecification).*

*(ii) If $C$ contains a point satisfying $J = 0$ and $s_e = \hat{p}$, then $\mathcal{P} = [0, 1]$.*

*(iii) For two candidates $\pi_1, \pi_2$ with observations $\hat{p}_1, \hat{p}_2$, write $\mathcal{P}_i = \mathcal{P}(\hat{p}_i, C_i)$.*

*(a)* ***Candidate-specific independent channels.*** *Suppose $(s_e^{(i)}, s_p^{(i)}) \in C_i$ and $\mathcal{P}_1, \mathcal{P}_2$ are nonempty and compact. Then $p_1 > p_2$ is uniformly identified if and only if*

$$\inf \mathcal{P}_1 > \sup \mathcal{P}_2,$$

*and symmetrically for $p_1 < p_2$. The strict ranking is not identified when*

$$\inf \mathcal{P}_1 \leq \sup \mathcal{P}_2 \quad \textit{and} \quad \sup \mathcal{P}_1 \geq \inf \mathcal{P}_2.$$

*(b)* ***Common channel.*** *Suppose a common channel $(s_e, s_p) \in C$ is shared by both candidates. Define the joint feasible set*

$$C_{12}^{\text{feas}} = \left\{ (s_e, s_p) \in C : J \neq 0,\ \frac{\hat{p}_i + s_p - 1}{J} \in [0, 1] \textit{ for } i = 1, 2 \right\} \cup \left\{ (s_e, s_p) \in C : J = 0,\ s_e = \hat{p}_1 = \hat{p}_2 \right\}, \tag{8}$$

*and assume $C_{12}^{\text{feas}} \neq \varnothing$; otherwise, the common-channel model is inconsistent with the observations.*
*If $\hat{p}_1 \neq \hat{p}_2$, then $C_{12}^{\text{feas}}$ contains only points with $J \neq 0$, and*

$$p_1 - p_2 = \frac{\hat{p}_1 - \hat{p}_2}{J}.$$

*Hence, the ranking agrees with that of $\hat{p}$ if $J > 0$ throughout $C_{12}^{\text{feas}}$, is reversed if $J < 0$ throughout the set, and is not identified if both signs of $J$ occur.*
*If $\hat{p}_1 = \hat{p}_2$, then $p_1 = p_2$ is identified when $C_{12}^{\text{feas}}$ contains only points with $J \neq 0$. If it contains a point with $J = 0$, then $p_1$ and $p_2$ are arbitrary and independent, so the ranking is entirely unidentified.*

*Proof.* (i) and (ii) follow from the unique solvability of the linear fractional relation when $J \neq 0$ and from the fact that $\hat{p} = s_e$ holds independently of $p$ when $J = 0$. For (iii)(a), sufficiency follows because separation implies a constant ranking over all consistent pairs; necessity follows because, if $\inf \mathcal{P}_1 \leq \sup \mathcal{P}_2$, compactness yields a consistent pair with $p_1 \leq p_2$. (iii)(b) follows from the expression for the difference and from the fact that, at points with $J = 0$, $\hat{p}_i = s_e$ does not depend on $p_i$. □

**Remark 1** (Computation)**.** *Only when $C$ is a rectangle on which the sign of $J$ is constant and bounded away from zero do the monotonicity relations $\partial p/\partial s_e = -p/J$ and $\partial p/\partial s_p = (1 - p)/J$ imply that the bounds of $\mathcal{P}$ are attained at the vertices of the rectangle. When $C$ crosses $J = 0$ or the truncation to $[0, 1]$ is active, the defining expression of the identified set must be handled directly.*

# C ResGap: Implementation and Evaluation Details

## C.1 Algorithm and Fine-Tuning Settings

Algorithm 1 summarizes the computation of ResGap. The same adaptation procedure is applied separately to the real dataset $D$ and the regenerated dataset $\tilde{D}$. The computation is repeated for three fine-tuning seeds and averaged.

**Data.** The real dataset $D$ is built from all 500 demonstrations of LIBERO-Object, that is, the 450 training demonstrations and the 50 held-out demonstrations. $D$ therefore includes the 50 held-out demonstrations used to compute the validation-loss baseline. The regenerated dataset $\tilde{D}$ is built from 100 training demonstrations, the first 10 of each task in file-name order, and contains no held-out demonstration. Each of the 100 is regenerated by $W$ from its first frame and its action sequence at $256^2$ resolution over its full length without segmentation. A training sample is an image–action pair drawn at a random time step of a random episode, with the action chunk of 16 steps $\times$ 7 dimensions at the stored LIBERO action scale in $[-1, 1]$, without further normalization. Images are downsampled from $256^2$ to $128^2$ and normalized to $[-1, 1]$. The two datasets differ in the size and composition of their episode pools.

**Algorithm 1** ResGap for one trained candidate

**Require:** trained candidate $\pi_{\theta_0}$; real dataset $D$ of image–action pairs from the task demonstrations; regenerated dataset $\tilde{D}$ (Section 5); loss $L$; steps $T_1, T_2$; learning rates $\eta_1, \eta_2$; seeds $\mathcal{S}$
**Ensure:** $\mathrm{ResGap}(\pi)$

```
for s ∈ S do
    for D' ∈ {D, D̃} do
        A ← A_0 (identity), θ ← θ_0
        train A for T_1 steps at η_1 on L(θ_0, A; D')  with θ frozen        ▷ stage 1
        train (θ, A) for T_2 steps at η_2 on L(θ, A; D') from (θ_0, Â)      ▷ stage 2
        δ_s(D') ← ‖θ̂ − θ_0‖_F / ‖θ_0‖_F                                    ▷ change of θ only
    end for
    ResGap_s ← δ_s(D̃) − δ_s(D)
end for
return (1/|S|) Σ_{s∈S} ResGap_s
```

**Adapter.** The adapter is residual, $A(x) = x + g(x)$, where $g$ is Conv2d($3 \to 16$, kernel $3 \times 3$, padding 1), ReLU, Conv2d($16 \to 3$, kernel $3 \times 3$, padding 1). The weights and biases of the last layer are initialized to zero, so $A_0$ is the identity. The adapter is applied to the image only; the gripper state and the language input are unchanged.

**Fine-tuning.** Both stages use the L1 loss on the action chunk, AdamW with the default weight decay of $0.01$, batch size 64, and $T_1 = T_2 = 300$ steps. Stage 1 trains only the adapter at $\eta_1 = 10^{-3}$. Stage 2 trains all parameters of the candidate, including its ResNet-18 visual encoder, together with the adapter in a single parameter group at $\eta_2 = 10^{-4}$. The change $\|\hat{\theta} - \theta_0\|_F / \|\theta_0\|_F$ is measured on the candidate's parameters only. The seeds are $\mathcal{S} = \{0, 1, 2\}$.

### C.2 Checkpoint Screening Evaluation

**Checkpoints.** The 17 ACT checkpoints come from four training series: the original run using 45 demonstrations per task, and three independently sampled runs using 5, 15, and 45 demonstrations per task. Each checkpoint is identified by its training series, seed, and number of training epochs. The ground truth is the simulator success rate $p$.

**Leave-one-series-out evaluation.** For each of the four series, we fit a linear regression with intercept from each predictor to $p$ using checkpoints from the other three series and apply it to the held-out series. Predictions are clipped to $[0, 1]$. ResGap is averaged over the three fine-tuning seeds described in Appendix C.1. We report Spearman rank correlation and MAE over the pooled out-of-fold predictions across all 17 checkpoints. Validation loss, number of demonstrations, training epochs, and the two ResGap terms, $\|u\|_F / \|\theta_0\|_F$ and $\|v\|_F / \|\theta_0\|_F$ are evaluated in the same way. The world-model evaluator is not regressed: its score $\hat{p}$ is used directly.

**Bootstrap intervals.** Table 2 reports percentile 95% bootstrap intervals from $B = 2000$ stratified resamples. Checkpoints are resampled with replacement within each training series, preserving the number of checkpoints in each series. For each resample, the leave-one-series-out evaluation is rerun and the metrics are computed from the pooled out-of-fold predictions. For the world-model evaluator, the metrics are recomputed directly from its scores on the same resample without regression.

**Out-of-series prediction.** To test transfer to new checkpoints, we fit a single regression on all 17 checkpoints, $p_{\mathrm{pred}} = 172.15\,\mathrm{ResGap} - 0.2606$, and apply it to four checkpoints from two training series not included in the 17-checkpoint analysis: two epochs from a run using 30 demonstrations per task (seed 5) and two from a run with 10 demonstrations per task (seed 6). Table 5 reports the predictions. Three of the four predictions are within $0.05$ of the measured success rate, whereas the checkpoint with $p = 0.037$ is predicted at $0.469$, accounting for most of the MAE of $0.130$.

Table 5: Out-of-series prediction with the regression fit on all 17 checkpoints. Predictions are clipped to $[0, 1]$.

| Series | Epochs | Predicted $p_{\text{pred}}$ | Measured $p$ | \|Error\| |
|---|---|---|---|---|
| 30 demos/task, seed 5 | 16 | 0.129 | 0.080 | 0.049 |
| 30 demos/task, seed 5 | 60 | 0.469 | 0.037 | 0.432 |
| 10 demos/task, seed 6 | 16 | 0.000 | 0.000 | 0.000 |
| 10 demos/task, seed 6 | 60 | 0.211 | 0.173 | 0.038 |

**Matched-pool variants.** The primary ResGap computation uses all 500 real demonstrations for $D$ and 100 regenerated demonstrations for $\tilde{D}$. To assess sensitivity to this difference in episode pools, we recompute ResGap using matched real and regenerated pools. When both sides use the same 100 demonstrations, the leave-one-series-out rank correlation decreases from $0.95$ to $0.68$ and the MAE increases from $0.088$ to $0.194$. Using matched pools of 450 training demonstrations gives a rank correlation of $0.88$ and an MAE of $0.120$. Thus, the original result is partly sensitive to the construction of the real and regenerated pools. We therefore treat ResGap as an empirical screening heuristic rather than interpreting it as an isolated measure of world-model regeneration error.

## D Controlled Environment: Details

**Environment.** The environment is a two-dimensional plane with horizon $H = 12$. The hand and the object start at $(0, 0)$, and the goal is $G = (1, 1)$. The hand command is

$$a_t = \text{clip}(2(G - h_t) + \xi_t, -1, 1)\,, \qquad \xi_t \sim \mathcal{N}(0, 0.10^2 I), \tag{9}$$

and the hand evolves as

$$h_{t+1} = h_t + 0.25a_t + \zeta_t, \qquad \zeta_t \sim \mathcal{N}(0, 0.01^2 I). \tag{10}$$

The bottleneck of Section 6.1 is the branch step $t = 6$, at which the hand is at approximately $(0.969, 0.969)$ without noise. An episode succeeds when $\|\text{obj} - G\| < 0.35$ at the final step $t = 12$. The object follows the hand with independent $\mathcal{N}(0, 0.01^2 I)$ noise while grasped. After release, it moves by $-0.30$ per step in $y$, with independent $\mathcal{N}(0, 0.01^2 I)$ dynamics noise; there is no floor, so a release at $t = 6$ ends at $y \approx -0.83$. The demonstration policy keeps the grasp at the branch step with probability $1 - \varepsilon$ and releases with probability $\varepsilon$.

**Candidates.** The four candidates are: a policy that never releases; the same policy with action-command noise increased from $0.10$ to $0.65$; a policy that always releases at $t = 6$; and a policy that always releases at $t = 10$. The true ranking is fixed, and the two releasing candidates have a success rate of zero. The true success rate of each candidate is measured over 200 episodes in the environment.

**World model.** The world model is an MLP with two hidden layers of 64 units that predicts the state residual from the current state and action. It is trained with Adam at learning rate $10^{-3}$ for 3000 steps with batch size 256 on the transitions from $n$ demonstrations. The transitions are pooled across demonstrations and randomly split into 80% for training and 20% for held-out evaluation. The held-out loss is the mean squared error on the held-out transitions. For each $\varepsilon \in \{0, 0.01, 0.05, 0.2\}$ and $n \in \{50, 200, 1000\}$, 20 random seeds are used, giving 240 runs.

**Measurements.** Each candidate is rolled out in the world model for 50 episodes, and its score is the fraction of predicted endpoints inside the goal radius, so that judge error is excluded. The counterfactual endpoint error is the absolute difference between the mean final object $y$-position over 50 world-model rollouts and a fixed reference mean over 200 environment rollouts, under a policy forced to release at the branch step. The misranking rate is the fraction of seeds for which the ordered top-two ranking induced by the world-model scores differs from the true ordered top two. Confidence intervals in Figure 2(c) are Clopper–Pearson 95% intervals over the 20 seeds. Ties are resolved by the fixed candidate order, which matches the true ordering and therefore biases the reported misranking rate downward.

Table 6: Prior evaluator designs and evaluator variations audited in the same setting. Correlation with $p$ is Pearson over the 24 ACT and Diffusion Policy candidates unless stated. "No" means that the score did not follow the consequence.

| | Design | Corr. with $p$ | Intervention | Grasp delay |
|---|---|---|---|---|
| E.1 | VLM progress reward (WorldGym) | $+0.59$ | No | No |
| E.2 | FID ranking (WorldEval) | $+0.55$ | No | No |
| E.3 | Forward–inverse consistency (SC3-Eval) | — | Detects | — |
| E.4 | World model with failure demonstrations | — | No | No |

Table 7: VLM progress-reward results.

| Analysis | Quantity | Result |
|---|---|---|
| Across 24 ACT + Diffusion Policy candidates | Pearson $r(\hat{r}, p)$ | $+0.59$ |
| | Spearman $\rho(\hat{r}, p)$ | $+0.71$ |
| | Pearson $r(\hat{r}, d)$ | $-0.76$ |
| | Spearman $\rho(\hat{r}, d)$ | $-0.80$ |
| | Linear $R^2$ from $d$ to $\hat{r}$ | 0.57 |
| Grasp delay | Simulator success $p$ | $0.91 \rightarrow 0.37$ |
| | Progress score $\hat{r}$ | $0.58 \rightarrow 0.60$ |
| Grasp-release intervention | Mean progress change (0–10 scale) | $+0.01$ |

# E Alternative Evaluator Audits and Model Variations

We examine three alternative evaluation designs and one model-level remedy in the same LIBERO-Object setting as the main audit. Where applicable, we apply the same grasp-release intervention and grasp-delay perturbation. Table 6 summarizes the results.

## E.1 VLM Progress Reward

WorldGym uses a VLM as a reward model (Quevedo et al., 2026). We use the same Qwen2.5-VL judge and eight uniformly sampled frames as in the binary evaluator, but ask the judge to rate task progress with the following prompt:

> *These are frames of a robot manipulation episode. Instruction: "{task}". Rate the task progress achieved by the end on a scale of 0 to 10, where 0 = no progress, 5 = object grasped and being transported, 10 = object fully placed in the basket. Answer with only a single integer.*

We divide the rating by 10 and average it over generated episodes to obtain the progress score $\hat{r}$, and the results are shown in Table 7.

The progress score is strongly associated with demonstration proximity, but it does not decrease when the grasp delay lowers simulator success or when the grasp-release intervention makes the modified action sequences fail. As a result, replacing the binary success judgment with a graded progress score does not resolve the observed insensitivity to action consequences.

## E.2 FID-Based Ranking

WorldEval ranks candidates by the FID between generated rollouts and reference videos (Li et al., 2025b). We compute the FID of each candidate's generated rollouts against the training demonstrations. As shown in Table 8, under the grasp-release intervention, the FID of the regenerated videos differs from that of the original videos by at most $0.5$. The grasp delay likewise leaves FID nearly unchanged, whereas perturbations that substantially alter the trajectory increase it. Thus, FID is sensitive to some large trajectory changes, but does not necessarily track changes in task outcome.

Table 8: FID-based evaluation results.

**(a) Across-candidate associations**

| Candidates | Quantity | Pearson $r$ | Spearman $\rho$ |
|---|---|---|---|
| All 31 | $-$FID vs. $\hat{p}$ | $+0.83$ | $+0.94$ |
| | FID vs. $d$ | $+0.64$ | $+0.84$ |
| | $-$FID vs. $p$ | $-0.19$ | $+0.19$ |
| ACT + Diffusion Policy | $-$FID vs. $p$ | $+0.55$ | $+0.80$ |

**(b) Under policy perturbations**

| Condition | FID |
|---|---|
| Base | 32.3 |
| Grasp delay | 32.6 |
| Action scale 0.6 | 39.9 |
| Observation delay 2 | 67.3 |

### E.3 Forward–Inverse Consistency

SC3-Eval checks whether an inverse dynamics model recovers the commanded actions from the generated video (Tseng et al., 2026). Our inverse dynamics model is a ResNet-18 initialized on ImageNet, with the first convolution is modified to accept two $128 \times 128$ frames four simulator steps apart, concatenated along channels, and the final layer replaced to output the mean arm action over the four steps (6 dimensions) and a logit for the sign of the gripper command. It is trained on random frame pairs from the 450 training demonstrations for 3000 steps with batch size 128 and AdamW at learning rate $3 \times 10^{-4}$, with the L1 loss on the arm action plus 0.2 times the binary cross-entropy on the gripper sign.

Table 9: Forward–inverse consistency under the grasp-release intervention.

| Data | Arm error | Gripper agreement |
|---|---|---|
| Held-out demonstrations | 0.048 | 0.97 |
| Original intervention pairs | 0.054 | 0.92 |
| Regenerated intervention pairs | 0.054 | 0.71 |

Table 9 shows the result. It does not recover the outcome of the modified action: the video still shows a successful transport, and the check indicates only that the gripper command is not followed.

### E.4 World Model Fine-Tuned with Failure Demonstrations

dWorldEval argues that world models trained primarily on successful demonstrations can render failure actions as successful outcomes, and that adding failure trajectories alone does not fully resolve this issue (Li et al., 2026). We fine-tune the main world model with 500 failure episodes added to the demonstrations. The failure episodes are simulator renderings of failed closed-loop episodes of the candidate policies, 50 per task sampled at random (seed 0) from the pooled trajectories of all candidates, with the gripper state synthesized from the commanded actions by a calibrated model.

Table 10: Evaluation results before and after augmenting world-model training with failure demonstrations.

| Quantity | Original WM | + Failure demonstrations |
|---|---|---|
| Intervention judgment-change rate | 0.050 | 0.031 |
| Unmodified control score | 0.844 | 0.525 |
| Base checkpoint score | 0.74 | 0.38 |
| Grasp-delay score | 0.84 | 0.41 |

The results are shown in Table 10. Adding failure demonstrations substantially changes the evaluator scores but does not restore sensitivity to the grasp-release intervention or the grasp delay. The added failure episodes do not contain grasp release during transport, so the distinguishing action remains unsupported.

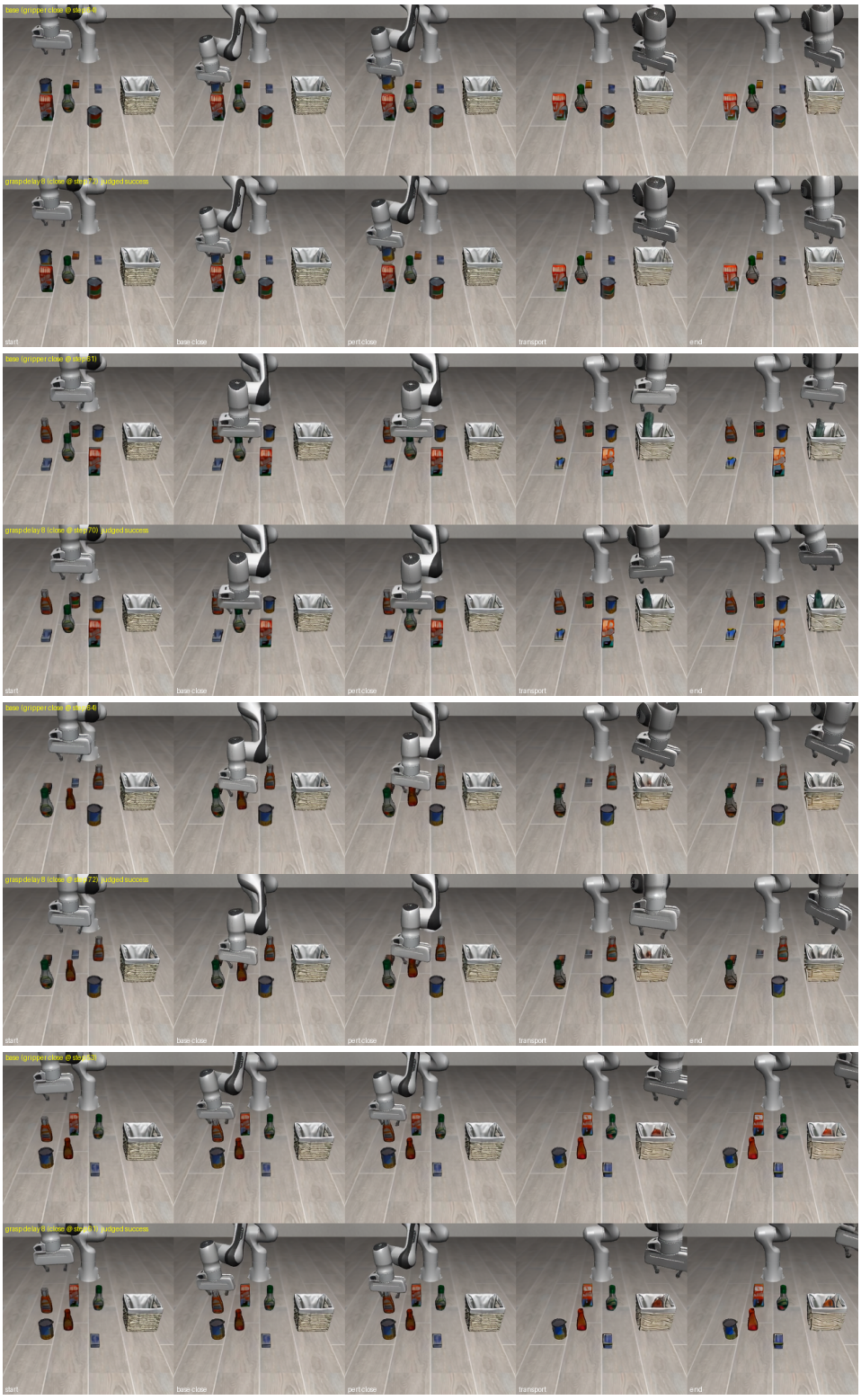

Figure 4: Examples under the grasp-delay perturbation that are judged successful. For each example, the top row shows the base checkpoint and the bottom row shows the same checkpoint with the grasp delayed by eight steps, from the same initial state. Frames from left to right show the start, the base grasp step, the delayed grasp step, transport, and the final frame.



Figure 5: Examples under the grasp-delay perturbation that are judged unsuccessful. The layout follows Figure 4.