CORRELATION IS NOT SELECTION: VALIDATING AUTOMATIC EVALUATORS FOR ROBOT POLICIES

Anonymous authors
Paper under double-blind review

ABSTRACT

Testing robot policies on real hardware is slow, so automatic evaluators, such as reward models, vision-language judges, and world models, are increasingly used to choose which policy to deploy. What matters is the evaluator's top pick, yet evaluators are validated by how well their scores rank-correlate with success measured on real robots, over three to eight policies. We investigate whether such evaluators pick the right policy from recorded robot trials. We first compile an extensive set of 82 published evaluator–reality correlations from nine papers. Most of these rest on too few policies to support a general ranking claim: of the 36 computed over four or more policies, 25 have a 95% interval that includes zero. We then audit eight open-weight evaluator configurations on the same 5,106 recorded episodes of seven robot-arm policies, against one set of double-blind human comparisons from the RoboArena benchmark. For each evaluator, we measure which policy it picks, how much worse that pick is than the humans' best, and how stable the pick is when the data is resampled. Three conclusions follow from these exploratory analyses. First, rank correlation does not tell which policy an evaluator will pick. Evaluators with the same correlation pick different policies, and they pick the humans' best policy in 16% to 80% of resamples, although these differences are not statistically established. Second, when evaluators miss the humans' best policy, they usually pick one almost as good, which the best policy would beat only slightly more than half the time. The exception, a score of how well a video's frames match the text instruction, picks the worst policy. Third, the best policy itself depends on whose preferences count and which policies compete. Counting each lab or human rater equally changes it, and adding one stronger policy drops the best evaluator's agreement to 20%. Evaluators should therefore be validated by the decision they support, against a named population and candidate set, not by correlation alone. We release the audit code, the frozen human ranking, and a report-card script that applies this audit to any new evaluator.

1 INTRODUCTION

Testing a robot policy on real hardware is slow and expensive. Automatic evaluators provide a favorable alternative. Reward models and vision-language model (VLM)-based judges score episode videos, and world models or real-to-sim simulators run episodes without a robot. Eleven such systems for choosing which policy to deploy have been presented in a span of thirteen months (Li et al., 2025; Quevedo et al., 2026; Guo et al., 2026; Jangir et al., 2026; Jain et al., 2026; Tseng et al., 2026; Jeon et al., 2026; Li et al., 2026; Ma et al., 2026; GigaWorld Team et al., 2026; Lee et al., 2026).

Most automatic evaluators are validated by rank correlation with real outcomes, such as success rates measured on real robots, over three to eight policies or checkpoints (saved training snapshots). A common rank correlation is Kendall's τ, which measures the share of policy pairs the evaluator orders correctly minus the share it orders wrongly, from -1 to $+1$. What is deployed is the evaluator's top-ranked policy. A high rank correlation does not say whether that policy is the best, a close second, or a poor one. Swapping only the top two of seven policies still keeps τ at $19/21 = 0.905$.

Judging whether an evaluator picks the right policy is hard for three reasons. First, the correlations published for these evaluators rest on few policies. We compiled all 82 correlations between evaluator scores and real outcomes reported in nine evaluator papers (Li et al., 2025; Quevedo et al., 2026;

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

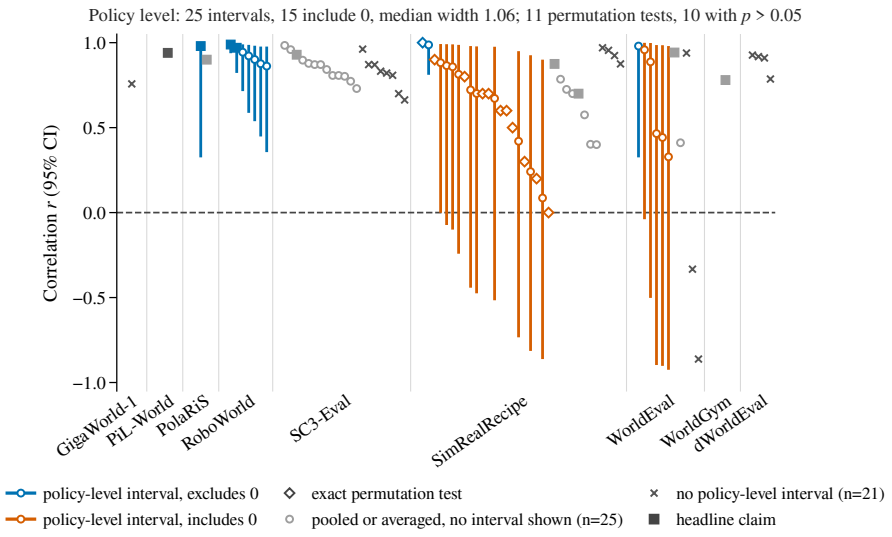

Figure 1: Every correlation with real outcomes recovered from nine evaluator papers, by paper (filled squares: values the papers highlight). Colored bars are 95% intervals for correlations over independent policies, blue if the interval excludes zero and orange otherwise. Diamonds are exact permutation tests from printed rankings. Grey circles pool over cells (one policy on one task) and get no interval. Crosses have no policy-level interval.

Jain et al., 2026; Tseng et al., 2026; Jeon et al., 2026; Li et al., 2026; Ma et al., 2026; GigaWorld Team et al., 2026; Wang et al., 2026a). Only 36 of them are computed across four or more separate policies, the kind that speaks to ranking policies, and for 25 of these 36, zero correlation cannot be ruled out, i.e., their 95% intervals include zero or their permutation tests are not significant (Figure 1, Appendix I). One study supplies 22 of the 36, and these intervals concern generalization to new policies, not the order of the candidates each paper tested. Second, the best policy itself is uncertain when the top policies are close. Our *human reference*, the policy ranking we compare evaluators with, is implied by double-blind human comparisons from RoboArena (Atreya et al., 2025). Its first policy beats its second with a fitted probability of only 0.544. Third, the best policy depends on whose judgments define it. RoboArena's comparisons come from many raters at many sites, but a single rater made 54% of the 2,553 comparisons behind our human reference (each comparison, or *session*, pits two policies on one task), so the human reference largely reflects one person's judgments.

We therefore test the evaluators' decisions directly on recorded robot trials. For each evaluator, we ask four questions: (i) which policy it picks, (ii) how much worse that pick is than the humans' best policy (the gap in their strengths fitted to the human comparisons), (iii) how stable the pick is when the data is resampled, and (iv) whether these answers hold for another population of human raters, another set of candidate policies, and sessions the evaluator did not see. We score 5,106 real episodes (single task attempts) of seven policies for the DROID robot arm (Khazatsky et al., 2024) with eight open-weight evaluator configurations: two reward models, four VLM judges, LLM-as-a-Verifier, which reads token probabilities, and an instruction–frame similarity score. Every evaluator sees the same episodes from the same camera, and one procedure turns each evaluator's scores into a ranking. To our knowledge, this is the first decision-level audit that compares eight evaluators under identical inputs against one human reference.

These exploratory analyses support four conclusions, each illustrated by a case from our audit:

(i) *A high correlation does not guarantee that the evaluator picks the best policy.* Qwen2.5-VL-72B and Qwen2.5-VL-7B have the same τ with the human ranking ($+0.714$), yet the 72B picks the humans' best policy and the 7B picks the runner-up.

(ii) *Missed picks are usually almost as good, with one exception.* When an evaluator picks the runner-up, the humans' best policy would beat it in only about 54% of the head-to-head comparisons. In contrast, SigLIP2, which scores a video by how closely its frames match the text instruction, ranks the policies almost in reverse and picks the worst one.

(iii) *Among close competitors, evaluators beat chance only modestly.* Once the one far worse candidate is set aside, the remaining six are close in quality, and the evaluators' picks, except SigLIP2's, come only slightly closer to the humans' best policy than a random pick does (a strength gap of 0.14 to 0.20 against 0.25, with 200 sessions to choose from).

(iv) *The best policy depends on whose preferences count and which policies compete.* If each lab counts equally, instead of letting the busiest human rater dominate, a different policy becomes the humans' best (the one ranked fourth in the pooled ranking). When a newer, stronger policy is added to the candidates, two evaluators that had picked the best policy, Qwen2.5-VL-72B and LLM-as-a-Verifier, keep picking the old one and miss the new best.

Together with the findings above, this work makes three contributions:

1. **A decision-level audit for automatic evaluators.** A protocol that validates an evaluator by the decision it supports. It measures which policy an evaluator picks, what the pick costs, and how stable it is, each against a named reference population and candidate set, with checks on held-out sessions, reweighted populations, and larger candidate sets (Section 3).
2. **An empirical study of published validations and of eight evaluators.** We reconstruct all 82 published evaluator–reality correlations (Appendix I) and audit eight open-weight evaluator configurations against RoboArena's human comparisons, which yields the conclusions above. The study also shows that equally valid implementation changes alter up to a quarter of an evaluator's verdicts but move its selection far less (Section 4).
3. **An open benchmark and toolkit for decision-level evaluation.** We release the frozen human reference with its bootstrap draws, per-episode scores and raw outputs of all eight configurations, the literature reconstruction, the full audit suite, and a report-card tool that turns any new evaluator's per-episode scores into the complete audit with one command on a CPU (Appendix L).

The rest of the paper is organized as follows. Section 2 presents related work, Section 3 defines the human reference and the measures, Section 4 reports the findings, and Section 5 gives the report card and the limitations. Full statistics and the literature reconstruction are provided in the appendices.

2 Related work

Automatic evaluators for robot policies. Automatic evaluators that score robot behavior without a real robot fall into three categories: (1) *World models* generate the episode, and human raters or a model judge success (Li et al., 2025; Quevedo et al., 2026; Guo et al., 2026; Tseng et al., 2026; Jeon et al., 2026; Li et al., 2026; Ma et al., 2026; GigaWorld Team et al., 2026); (2) *Reward models* score a real episode directly (Lee et al., 2026; Wang et al., 2026c); (3) *Language models and VLMs* are prompted as judges (Zheng et al., 2023), read a continuous score from token probabilities (Kwok et al., 2026), or supply a similarity reward (Rocamonde et al., 2024; Sontakke et al., 2023). R2S-Eval (Wang et al., 2026b) has VLMs compare pairs of episodes and combines their preferences with the Bradley–Terry model, which fits one strength per policy from pairwise wins. We use the same model to turn each evaluator's scores into a ranking. Most of these systems are validated by correlation with real outcomes, and we directly test the policy choice that such correlations are meant to support.

Human ground truth. RoboArena (Atreya et al., 2025) runs distributed double-blind A/B evaluation on DROID (Khazatsky et al., 2024) and releases the raw comparisons, and ArmnetBench (Selvaraj et al., 2026) adds a second human-scored dataset. Tian et al. (2026) test three reward models against RoboArena human labels at the level of trajectory pairs. We move the audit to the level of policy selection and ask whose preferences the human reference encodes.

Auditing evaluators. In language modeling, LLM judges show documented biases (Panickssery et al., 2024; Wang et al., 2024) and face a proven limit as substitutes for human labels (Dorner et al., 2025). Audits in adjacent areas also report substantial disagreement between evaluators and humans (Bhat et al., 2026). Numerical nondeterminism, meaning run-to-run differences in floating-point results, also changes LLM outputs (Yuan et al., 2025; He & Thinking Machines Lab, 2025), most of all for long reasoning chains (Yuan et al., 2025). Embodied evaluators inherit these issues and add video decoding and frame sampling. SIMPLER (Li et al., 2024) measures rank violations between simulation and reality, while we measure the selection decision and its dependence on the reference population. Appendix K discusses further related work.

3 AUDITING POLICY SELECTION

An automatic evaluator is useful to a practitioner only if it selects the policy that human raters would select. Thus, in this work, we audit that decision directly.

3.1 THE REFERENCE RANKING AND ITS POPULATION

We use the RoboArena data dump (Atreya et al., 2025), which contains 3,284 double-blind pairwise comparisons over 9,589 episodes from 15 policies on the DROID platform (Khazatsky et al., 2024). Seven policies have at least 600 episodes each, one policy (`pi05_droid`) has 564, and the remaining seven have at most 60. Our human reference is fitted to the 2,553 sessions in which both compared policies belong to the seven with at least 600 episodes, a total of 5,106 episodes. All seven are vision-language-action models (VLMs that output robot actions) fine-tuned on DROID. Two start from the robot model π_0, and five start from the VLM PaliGemma and differ only in how they represent actions (Appendix B). We drop their shared suffix `_droid` and write `pg` for PaliGemma.

A Bradley–Terry model (Bradley & Terry, 1952) turns pairwise wins into one strength per policy. Each policy i gets a log-strength θ_i, and i beats j with probability $1/(1 + e^{-(\theta_i - \theta_j)})$. Ties count as half-wins. We fit it to the reference sessions and froze it before any evaluator ran. Its order is `pi0_fast` > `pg_diffusion` > `pg_vq` > `pg_fast_spec` > `pg_fast` > `pi0` > `pg_binning` (Appendix B).

Whose preference is the human reference? The human reference reflects the preferences of the RoboArena raters, on their mix of tasks, and these raters are not equally represented. One rater at one site supplies 1,370 of the 2,553 sessions (53.7%). That rater, that site, and that site's tasks always occur together, so their effects cannot be separated. A claim that an evaluator "recovers the human ranking" is therefore incomplete unless it says whose ranking it means.

We check each decision-level result in two ways. The first keeps the pooled reference and measures uncertainty with a bootstrap, which redraws the data many times, with replacement, to show how stable a result is. We run it with three *resampling units*, redrawing sessions, raters, or sites. The second changes the human reference itself with three *reweighted targets*, which count every site equally, count every rater equally, or drop the rater who dominates the pool (the *dominant contributor*). These targets can change which policy counts as best (Appendix B).

Protocol. We pre-registered the policy set, the aggregation, six hypotheses, and the decision rules before we scored the full set of episodes (Appendix J). Pre-registering means fixing the analysis plan before seeing the data. Winner agreement and regret, our main decision-level measures, were added after the registered analysis, so they are exploratory. Table 1 gives the status of every result.

3.2 HOW WE MEASURE AN EVALUATOR'S CHOICE

Each evaluator scores every episode. Within a session, its *induced preference* is the episode it scores higher, and equal scores tie. The same Bradley–Terry code turns these preferences into the evaluator's ranking. We measure three quantities with a paired bootstrap of 10,000 draws, which redraws the same sessions for the evaluator and the human reference. Each 95% percentile interval is the middle 95% of the draws. *Rank agreement* is Kendall's τ (Kendall, 1938). With seven policies, τ can move

Table 1: Status of each result. *Registered*: fixed before we scored all episodes. *Amended*: changed before we examined their outcome. *Exploratory*: added after we saw results.

registered	amended	exploratory
reference and policy set; induced-preference aggregation; τ with session bootstrap; H1 (evaluators disagree on the winner); H3 (7-policy resolution); H5 (benchmark error vs. ranking); contamination control	parse rule for tag-declaring judges; confound arms; noise floors on every judge	winner agreement and regret; rater, site, and dominant-contributor targets; expected-value readout; mean-score aggregation; inclusion thresholds; 72B and LLM-as-a-Verifier; human progress score; stacked re-scoring (a learned combination of evaluators); measured reruns; corrected literature table

only in steps of 0.095. In one bootstrap draw (marked *), let $\hat{\theta}^*_{E,i}$ and $\hat{\theta}^*_{H,i}$ be the log-strengths of policy i fit to the evaluator (E) and to the human reference (H), and let $i^*_E = \arg\max_i \hat{\theta}^*_{E,i}$ and $i^*_H = \arg\max_i \hat{\theta}^*_{H,i}$ be their top policies. Then

$$\mathrm{WA} = \Pr_*\left[\, i^*_E = i^*_H \,\right], \qquad R^* = \hat{\theta}^*_{H,i^*_H} - \hat{\theta}^*_{H,i^*_E}, \tag{1}$$

where $\Pr_*$ is the share of draws. *Winner agreement* WA is how often the evaluator's top policy is the reference's. *Selection regret* R^* is the reference log-strength given up by deploying the evaluator's pick instead of the reference's, and it is zero when they agree. We report its mean and its whole distribution, so we do not have to choose one tolerance margin. *Frozen-choice regret* fixes i^*_E at the evaluator's pick from all the data, the single policy a practitioner would deploy. *Held-out regret* fits $\hat{\theta}_H$ on sessions the selection did not use, so the pick is not scored on the data that chose it. Regret also has a direct reading as a probability: at regret Δ, the reference model predicts that its winner beats the evaluator's pick in a head-to-head comparison with probability $1/(1+e^{-\Delta})$, for example 0.55 at $\Delta = 0.2$. Neither measure is a calibrated probability of choosing the population-best policy in a future evaluation, and the 10,000 draws are not independent experiments. Bootstrap statements about empirical ranks can be inconsistent near ties (Hall & Miller, 2009), so a simulation checks our intervals (Appendix C).

3.3 Evaluators and controls

We audit six open-weight evaluators that cover the three ways an episode is scored today. Two are reward models, RoboReward-4B and RoboReward-8B (Lee et al., 2026). Three are VLM judges, namely Qwen2.5-VL-7B (Bai et al., 2025), InternVL3-8B (Zhu et al., 2025), and Cosmos-Reason2-8B (NVIDIA, 2025; Azzolini et al., 2025), that use chain-of-thought, which means they write out their reasoning before they give a score. The last evaluator, SigLIP2-B/16 (Tschannen et al., 2025), is a similarity score. It scores an episode by the mean cosine between the embeddings of the instruction and of each frame, a zero-shot reward (one that needs no training) (Rocamonde et al., 2024) and a per-frame form of RoboCLIP's instruction–video similarity (Sontakke et al., 2023). We later added a larger judge, Qwen2.5-VL-72B, and LLM-as-a-Verifier (Kwok et al., 2026) run on Qwen3.5-9B. All eight evaluators are listed in Table 2. Every evaluator scores each episode on its own, from the task instruction and the video of one exterior camera: the right camera when its video is available (4,461 of 5,106 episodes) and otherwise the left (645). It never sees the other episode of the session or the raters' notes. Appendix A gives the number of frames each model receives. Because the weights are open, we can fix each checkpoint, library version, and attention kernel (the GPU routine that computes attention), which proprietary API judges do not allow.

Four controls accompany every evaluator (Appendix H). The first is a clean cohort without the episodes in RoboReward's training data (Section 4.4). The second changes the readout, the rule that turns a model's answer into a score. The *expected-value readout* replaces a judge's one answer from 1 to 5 with the mean of the answers, weighted by the model's probabilities, so only the readout changes. The third re-runs each evaluator under an *equally valid configuration*, which changes only a kernel or library version, and the share of scores that change is its *noise floor*. The fourth consists of blind and confound arms, which remove the pixels or alter the frames. Finally, the *human progress score* passes the raters' own per-episode progress scores through the same pipeline.

4 Findings

Every evaluator was run on all 5,106 audit episodes. Two evaluators scored slightly fewer sessions, so Table 2 compares evaluators on the 2,498 sessions that all of them scored, the *matched cohort*. All evaluators are compared against the same human ranking. This choice matters for RoboReward-4B, which picks the reference winner instead of the runner-up on all 2,553 sessions. Results on each evaluator's *own sessions* (all the sessions it scored), full statistics, and compute are in Appendix C.

4.1 Rank agreement does not determine selection

Every evaluator except SigLIP2 ranks the policies in the human reference's direction. Their τ lie between $+0.619$ and $+0.810$, and every interval excludes zero. A one-sided exact permutation test against a random ranking also gives $p \le 0.035$ for each. Yet the τ intervals overlap almost completely, while the evaluators make different choices with different stability. Qwen2.5-VL-72B and its 7B sibling have the same τ ($+0.714$) but pick different policies. RoboReward-8B and LLM-as-a-Verifier have the same τ ($+0.810$, paired difference $\Delta\tau = 0.000$ $[-0.190, +0.476]$) and pick

Table 2: Rank agreement and selection on the matched cohort (2,498 sessions), with measures as in Section 3.2. τ with its 95% paired bootstrap interval. *Pick rank* is the reference rank of the evaluator's pick. Winner agreement under three resampling units and three reweighted targets, which reweight the evaluator too. The session column adds a nominal double-bootstrap 95% interval, and the first rows give the reference winner under each target and how often it holds. Regret: selection regret (upper) and frozen-choice regret (lower), session unit. Bold: highest winner agreement per target (no contrast is established, Appendix C).

			resampling unit			reference target			
evaluator	τ 95% CI	pick rank	session	rater	site	sites equal	raters equal	w/o dominant	regret res./frozen
reference winner holds in			pi0_fast 0.945	pi0_fast 0.690	pi0_fast 0.692	pg_fast_spec 0.449	pg_vq 0.530	pg_vq 0.536	
original roster									
RoboReward-4B	$+0.619$ $[+0.143, +0.810]$	2	0.325 [0.04,0.87]	0.426	0.491	0.318	0.252	0.215	0.131 0.182
RoboReward-8B	$+0.810$ $[+0.238, +0.905]$	1	0.419 [0.07,0.91]	0.461	0.489	**0.477**	0.266	0.273	0.097 0.002
Cosmos-Reason2-8B	$+0.619$ $[+0.238, +0.905]$	2	0.159 [0.01,0.67]	0.059	0.049	0.088	0.234	0.102	0.162 0.182
Qwen2.5-VL-7B	$+0.714$ $[+0.333, +0.905]$	2	0.299 [0.04,0.88]	0.199	0.219	0.240	0.067	0.193	0.122 0.182
InternVL3-8B	$+0.714$ $[+0.238, +0.905]$	2	0.169 [0.01,0.77]	0.224	0.244	0.222	0.228	**0.413**	0.158 0.182
SigLIP2-B/16	-0.714 $[-0.905, -0.238]$	7	0.000 [0.00,0.00]	0.002	0.002	0.005	0.082	0.001	3.125 3.125
added configurations									
Qwen2.5-VL-72B	$+0.714$ $[+0.429, +1.000]$	1	0.573 [0.13,0.94]	0.445	0.462	0.133	**0.303**	0.198	0.066 0.002
LLM-as-a-Verifier Qwen3.5-9B, 4 repeats	$+0.810$ $[+0.429, +1.000]$	1	**0.795** [0.20,0.98]	**0.616**	**0.571**	0.382	0.246	0.247	0.026 0.002
human progress	$+0.905$ $[+0.714, +1.000]$	1	0.967 [0.68,1.00]	0.858	0.833	0.725	0.715	0.633	0.001 0.002

the same policy, yet recover the winner in 0.419 and 0.795 of draws. Across evaluators with positive τ, winner agreement ranges from 0.159 for Cosmos to 0.795 for LLM-as-a-Verifier (Figure 2a). These differences are observed but not established. Each evaluator's own interval is wide (Table 2). A double bootstrap, which redraws the data again inside each bootstrap draw, puts the difference between RoboReward-8B and LLM-as-a-Verifier at $[-0.28, +0.71]$ and between LLM-as-a-Verifier and InternVL3 at $[-0.11, +0.86]$, so both include zero (Appendix C). A simulation in the same appendix shows that at our sample size the audit would almost never detect an evaluator that aims its errors at the two strongest policies, although this lowers its true winner agreement by 0.1 to 0.2.

Rank agreement and selection also respond differently to the decision rule (own sessions, Appendix E). Ranking policies by their mean score, instead of comparing the two episodes of each session, lowers τ for every positive-τ evaluator except Qwen2.5-VL-72B, yet raises winner agreement for four, for example from 0.591 to 0.889 for Qwen2.5-VL-72B (Table 22). The expected-value readout has the same split effect, nearly all of it from breaking ties (Table 21). Every interval for these changes includes zero, so neither rule is established as better.

4.2 Selection depends on whose preference the reference encodes

Under the three resampling units in Table 2, the reference winner remains `pi0_fast`. When each lab or each rater counts once, or when the dominant contributor is removed, the winner becomes `pg_fast_spec` or `pg_vq` (Table 2, first rows). None of these alternative winners holds in more than 0.54 of draws. Which evaluator agrees best with humans depends on the target (Figure 2b). Under session resampling, LLM-as-a-Verifier has the highest winner agreement (0.795). Under each reweighted target, its winner agreement falls to 0.25–0.38, and a different evaluator leads. RoboReward-8B leads when each site counts equally (0.477). Qwen2.5-VL-72B leads when each rater counts equally (0.303). InternVL3, one of the lowest under session resampling, leads when the dominant contributor is removed (0.413). Reweighting only the reference, with each selection

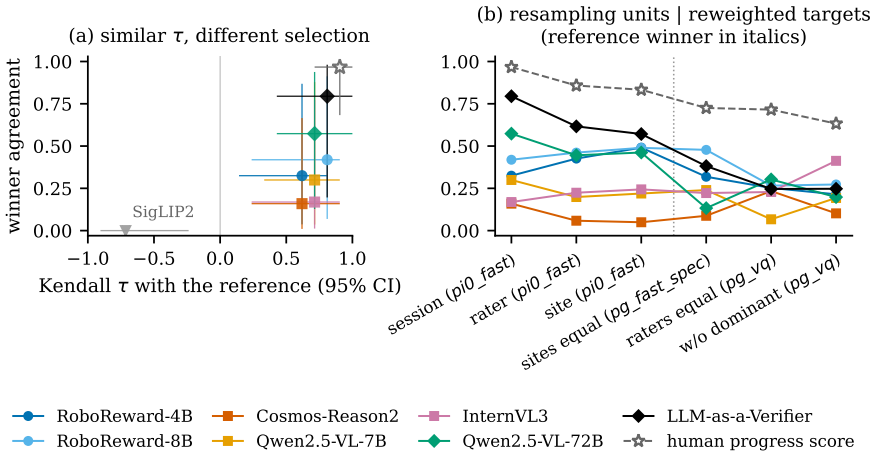

Figure 2: (a) Rank against winner agreement, matched cohort, nominal 95% intervals (winner agreement by double bootstrap). (b) Winner agreement under each resampling unit and reweighted target. The reference winner (italics) and the best-agreeing evaluator both change under the targets.

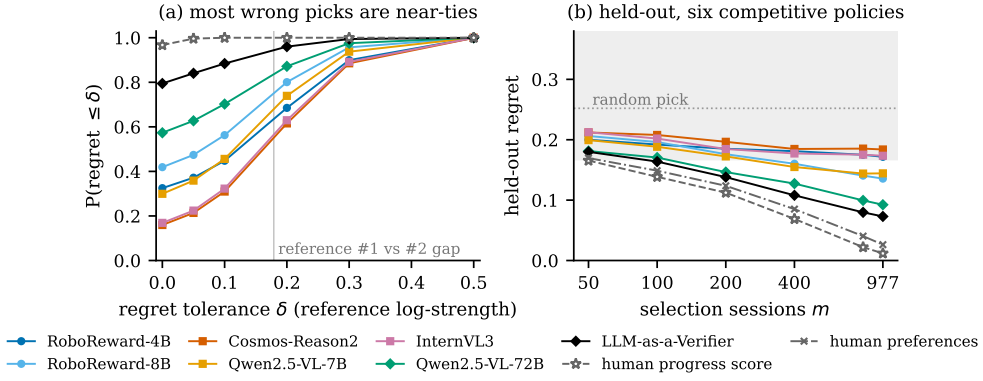

Figure 3: (a) Share of bootstrap draws in which the evaluator's pick gives up at most a tolerance δ of reference log-strength, session unit, matched cohort. The value at $\delta = 0$ is winner agreement. (b) Held-out regret among the six competitive policies (`pg_binning` removed): each pick is made on m sessions and scored against a reference fit on disjoint sessions. The dotted line and band are a random pick and its nominal 95% interval. SigLIP2 (0.28 to 0.44) is not shown.

rule fixed, lowers the human progress score's winner agreement as far as that of the evaluators that pick `pi0_fast`, so this drop measures how far the reference moves, not an evaluator failure (Appendix C). The reference carries a clear signal, because the human progress score keeps its winner agreement between 0.633 and 0.967 under all six resampling units and targets.

4.3 The cost of a wrong pick

Winner agreement counts every wrong pick the same way, whereas regret measures how much each one costs (Figure 3a). Every evaluator with positive τ picks either the reference winner or the runner-up, whose log-strength is 0.179 lower. Their mean regret is 0.026 to 0.162, and their pick is within 0.2 of the reference winner in 62% to 96% of draws. Under the reference's own model, the reference winner would beat these picks with probability 0.506 to 0.540 on average, close to a coin flip. These evaluators often miss the exact winner but pick a near-tie.

Frozen-choice regret scores the single policy each evaluator would deploy. RoboReward-8B, Qwen2.5-VL-72B, and LLM-as-a-Verifier pick the winner of the reference fit on all sessions, with a frozen-choice regret of 0.002. RoboReward-4B, Cosmos, Qwen2.5-VL-7B, and InternVL3 pick the runner-up, which costs 0.18. RoboReward-8B's winner agreement is low, yet its full-data pick is correct,

Table 3: InternVL3 re-run on all audit episodes under an equally valid attention kernel.

	episode scores	session outcomes	session reversals	adjacent rank exchanges	pick changed
changed	747 / 5,105	463 / 2,552	35 / 2,552	1	no
share	14.6%	18.1%	1.4%	—	—

so its procedure is unstable around a correct decision. On held-out regret, the positive-τ evaluators give up 0.05 to 0.19 over all seven policies, against 0.67 for a random pick, but most of that margin comes from avoiding the one far-worst policy. Among the six competitive policies the gain is modest (Figure 3b). With 200 selection sessions they give up 0.14 to 0.20, against 0.25 for a random pick. With the full half of the sessions, only LLM-as-a-Verifier and Qwen2.5-VL-72B clearly beat chance. We cannot establish that any evaluator does worse than a pick made from the raters' own preferences on the same sessions (Appendix C).

SigLIP2, the basic zero-shot reward of Rocamonde et al. (2024) applied to evaluation, behaves very differently. It ranks the policies in reverse ($\tau = -0.714$) and picks the reference's last policy in every draw, a regret of 3.12. Its regret stays large under every resampling unit and target (Appendix C). The confound arms show that the reversal comes from the later frames, because on the first frame alone SigLIP2's agreement with the reference is positive, although not significant (Appendix H). One explanation, consistent with this control but not identified by it, is that a policy that barely moves leaves the scene close to the instruction's description, whereas a policy that acts changes it.

4.4 Training on the human reference's labels

RoboReward is trained on RoboArena episodes with their human progress scores (Lee et al., 2026), including 1,322 of our 5,106 audit episodes. These scores come from the same raters, in the same sitting, as the reference preferences and agree with them in 94.1% of sessions, so the overlap leaks the reference's labels as well as its episodes. On the *clean cohort*, the 1,839 sessions with both episodes outside RoboReward's training data and a refit reference, Qwen2.5-VL-7B's pick moves to the reference winner and RoboReward overtakes LLM-as-a-Verifier under all three resampling units (Appendix H). The clean reference is more concentrated on one rater (73.8% against 53.7%), so this difference is not the effect of removing contamination.

4.5 Implementation changes move scores more than selections

Changing one equally valid implementation detail changes up to a quarter of an evaluator's verdicts (Appendix F). Under a torch-version change, Cosmos-Reason2, which uses chain-of-thought, changes its verdict on 23.1% of a 1,024-episode sample. InternVL3 changes 1.6% under the same change and 17.2% under an attention-kernel change, while the RoboReward models and SigLIP2 flip no preference. The long reasoning chain is the likely amplifier, as reported for text models (Yuan et al., 2025). On the full audit set, InternVL3's kernel change alters many scores and session outcomes, but nearly all are ties forming or breaking, and the pick does not change (Table 3). A noise floor should therefore name the changed setting and be followed through to the selection, because changed scores overstate changed selections.

4.6 More policies

The seven-policy reference follows a registered inclusion rule of at least 600 episodes per policy. Lowering the threshold to 300 episodes adds `pi05_droid` and 534 sessions, which we scored with all eight evaluators (Table 4). The new policy becomes the reference winner and holds that place in 0.838 of draws. The main findings still hold, i.e. the τ intervals of the positive evaluators overlap, and their winner agreement varies widely (Table 4). Three evaluators pick the new winner, but Qwen2.5-VL-72B and LLM-as-a-Verifier, which picked the winner among seven policies, stay with `pi0_fast`, now the runner-up. On its own sessions, LLM-as-a-Verifier's winner agreement falls from 0.827 to 0.197, although its τ remains high ($+0.786$) and its regret small (0.13). Winner agreement in one candidate set therefore says little about the next set, and the audit has to be rerun whenever the set changes. The extension also changes the rater population, because one rater supplies 68% of the added sessions. SigLIP2 again picks the last policy (regret 3.3), even though its τ interval now includes zero. With all fifteen policies, the reference is too thin to audit against (Appendix G).

Table 4: Three inclusion rules (minimum episodes per policy), own sessions. τ, winner agreement, and mean regret: ranges over the seven positive-τ evaluators, with SigLIP2 shown separately.

rule	poli-cies	sessions	reference winner (holds)	τ range	winner agreement	regret	SigLIP2 τ	SigLIP2 regret
≥ 600	7	2,553	`pi0_fast` (0.940)	$+0.62$–$+0.81$	0.21–0.83	0.02–0.15	-0.714	3.13
≥ 300	8	3,087	`pi05` (0.838)	$+0.50$–$+0.86$	0.10–0.58	0.08–0.22	-0.357	3.31
all	15	3,284	`dreaming_zebra` (0.667)	$+0.26$–$+0.71$	0.38–0.69	0.25–0.81	-0.219	2.01

5 Discussion

5.1 A report card for policy evaluators

Based on our findings, we recommend that every automatic evaluator paper report five things. Our script computes each of them from per-episode scores. *(1) Selection beside correlation.* Report the frozen pick and its regret, the held-out regret compared with a random pick among the competitive policies, and the winner agreement, which shows how stable the pick is. *(2) The reference population.* Report who produced the labels, how concentrated the labels are, and the result of at least one reweighting of raters or sites. *(3) The decision rule.* Report the readout and the aggregation. In our study, each of them raised winner agreement in some cases and lowered it in others. *(4) The noise floor at the decision level.* Report how many scores, session outcomes, and selections change under an equally valid configuration. *(5) Label-level contamination.* Report whether the evaluator was trained on the human reference's own labels. The repository holds a card for each evaluator in this paper (Appendix L).

5.2 Limitations

This paper is a case study of seven related policies on one embodiment (one type of robot) and one pool of human raters. Five of the seven share a PaliGemma backbone, making near-ties at the top likely. The decision-level analyses are exploratory, and a confirmatory replication would pre-register them on a second dataset, such as ArmnetBench (Selvaraj et al., 2026). With seven policies, τ moves in steps of 0.095, so we lead with selection and regret, whose resolution comes from 2,498 to 2,553 sessions. Rater, site, and task mix cannot be separated here, so the rater and site columns are sensitivity analyses, and population results describe the RoboArena pool, not individual raters.

We did not evaluate world models or proprietary API judges. Our results describe the configurations we ran, not whole model families. The 72B judge comes from one run. LLM-as-a-Verifier samples its outputs, so we ran it three times, and it picks the same policy in each run. Each evaluator sees one exterior camera, while the raters watch the robot itself, so the human progress score is a control that is aligned with the human reference and has information the evaluators lack.

6 Conclusion

Several results of this study were unexpected. In particular, two evaluators with the same rank correlation selected different policies. In addition, a plausible similarity reward ranked the policies in reverse. One human rater supplied 54% of the human reference's comparisons, and counting raters or labs equally changed the best policy. Moreover, adding one stronger policy made two evaluators that had picked the best miss the new best, with little change in rank correlation. Equally valid implementation changes altered up to a quarter of an evaluator's scores, yet in the one full rerun the pick was unchanged. In the literature, 25 of the 36 policy-level correlations cannot rule out zero.

Can automatic evaluators choose the correct robot policy? For a practitioner, our answer is mixed. Most evaluators we audited pick the humans' best policy or a near-tie that the best policy beats in barely more than half of comparisons. Among the competitive policies, however, their advantage over a random pick is modest. Whether they recover the exact winner is uncertain and depends on the reference population, the candidate set, and the decision rule, none of which rank correlation summarizes. Evaluators should therefore be validated at the decision level, against a named population and candidate set, and our report card makes this routine.

STATEMENT ON THE USE OF LARGE LANGUAGE MODELS

We used large language models for two purposes: to polish the writing of this manuscript, and to write part of the analysis and evaluation code. All research ideas, the experimental design, the pre-registration, and the interpretation of results are the authors' own. The authors reviewed the generated code, verified every number, table and figure against the archived run outputs, and checked every citation against its source. The authors take full responsibility for the content of this paper.

REPRODUCIBILITY STATEMENT

The registered protocol fixes the policy set, the aggregation rule, the hypotheses, the decision rules, the compliance threshold (the largest share of unparsed replies an evaluator may have), and the software environment; Appendix J gives the hypotheses, their outcomes, and what changed after registration, and Table 1 separates registered from exploratory results. Every checkpoint is pinned to an exact revision and the software environment is pinned (Appendix A); Appendix F reports what changes when a pin is varied. Raw model output is stored for every episode, so every score can be re-derived without a GPU. Every interval uses a fixed bootstrap seed. The frozen reference ranking with its bootstrap draws, the audit suite, the per-arm aggregation outputs, and the coefficient-level literature table are released in an anonymous repository: `https://anonymous.4open.science/r/evaluator-selection-audit/`.

ETHICS STATEMENT

This study uses the publicly released RoboArena data dump (Atreya et al., 2025), distributed under a permissive licence, which contains robot episode video and human preference labels tagged with each rater's name and site; we use these fields only to group sessions. We collected no new human-subject data. We report agreement between automatic evaluators and existing human labels, and we make no claim about the quality of any individual human annotator. One rater at one site supplies 53.7% of the sessions; because that rater and that site coincide, naming the site would identify a person, so the paper and the released bundle refer to sites and raters only by pseudonyms.

The paper is critical of published evaluation practice, so we state our standard for that criticism. We re-derive intervals from numbers those papers report themselves, we apply the same statistical treatment to our own results, and we report our own failed control and our own changes to the registered plan with the same prominence as our positive findings. We name evaluator systems because the claims we examine are public and attributable; we do not attribute error to individuals.

Automatic evaluators are attractive because real-robot evaluation is expensive, and that cost falls hardest on labs without hardware. An evaluator that is confidently wrong is worse for those labs than an honest interval, which is the practical reason for this audit.

REFERENCES

P. Atreya, K. Pertsch, T. Lee, et al. RoboArena: Distributed real-world evaluation of generalist robot policies. In *Conference on Robot Learning (CoRL)*, volume 305 of *PMLR*, pp. 336–364, 2025.

A. Azzolini, J. Bai, H. Brandon, et al. Cosmos-Reason1: From physical common sense to embodied reasoning. *arXiv preprint arXiv:2503.15558*, 2025.

S. Bai, K. Chen, X. Liu, et al. Qwen2.5-VL technical report. *arXiv preprint arXiv:2502.13923*, 2025.

L. Beyer, A. Steiner, A. S. Pinto, et al. PaliGemma: A versatile 3b VLM for transfer. *arXiv preprint arXiv:2407.07726*, 2024.

V. Bhat, J. Vaghasiya, M. A. Mohsin, et al. Benchmarking the benchmarks: A validity audit of tool-calling evaluation. *arXiv preprint arXiv:2607.02577*, 2026.

K. Black, N. Brown, D. Driess, et al. π_0: A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 2024.

X. Bouthillier, P. Delaunay, M. Bronzi, et al. Accounting for variance in machine learning benchmarks. In *Proceedings of Machine Learning and Systems (MLSys)*, volume 3, pp. 747–769, 2021.

R. A. Bradley and M. E. Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952.

A. Brohan, N. Brown, J. Carbajal, et al. RT-2: Vision-language-action models transfer web knowledge to robotic control. *arXiv preprint arXiv:2307.15818*, 2023.

J. Dodge, S. Gururangan, D. Card, et al. Show your work: Improved reporting of experimental results. In *Conference on Empirical Methods in Natural Language Processing and International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019.

F. E. Dorner, V. Y. Nastl, and M. Hardt. Limits to scalable evaluation at the frontier: LLM as judge won't beat twice the data. In *International Conference on Learning Representations (ICLR)*, 2025.

E. C. Fieller, H. O. Hartley, and E. S. Pearson. Tests for rank correlation coefficients. I. *Biometrika*, 44(3/4):470–481, 1957. doi: 10.2307/2332878.

GigaWorld Team, A. Ma, B. Wang, et al. GigaWorld-1: A roadmap to build world models for robot policy evaluation. *arXiv preprint arXiv:2607.02642*, 2026.

Y. Guo, L. X. Shi, J. Chen, et al. Ctrl-World: A controllable generative world model for robot manipulation. In *International Conference on Learning Representations (ICLR)*, 2026.

P. Hall and H. Miller. Using the bootstrap to quantify the authority of an empirical ranking. *The Annals of Statistics*, 37(6B):3929–3959, 2009. URL `https://arxiv.org/abs/0911.3749`.

H. He and Thinking Machines Lab. Defeating nondeterminism in LLM inference. `https://thinkingmachines.ai/blog/defeating-nondeterminism-in-llm-inference/`, 2025. Blog post, 10 September 2025.

A. Jain, M. Zhang, K. Arora, et al. PolaRiS: Scalable real-to-sim evaluations for generalist robot policies. In *Robotics: Science and Systems (RSS)*, 2026.

Y. Jangir, Y. Zhang, K. Yamazaki, et al. RobotArena ∞: Scalable robot benchmarking via real-to-sim translation. In *International Conference on Learning Representations (ICLR)*, 2026.

B. Jeon, S. Ye, J. Doo, et al. RoboWorld: Fast and reliable neural simulators for generalist robot policy evaluation. *arXiv preprint arXiv:2607.01060*, 2026.

M. G. Kendall. A new measure of rank correlation. *Biometrika*, 30(1–2):81–93, 1938.

A. Khazatsky, K. Pertsch, S. Nair, et al. DROID: A large-scale in-the-wild robot manipulation dataset. In *Robotics: Science and Systems (RSS)*, 2024. arXiv:2403.12945.

M. J. Kim, K. Pertsch, S. Karamcheti, et al. OpenVLA: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024.

J. Kwok, S. Li, P. Atreya, et al. LLM-as-a-Verifier: A general-purpose verification framework. *arXiv preprint arXiv:2607.05391*, 2026.

T. Lee, A. Wagenmaker, K. Pertsch, et al. RoboReward: General-purpose vision-language reward models for robotics. *arXiv preprint arXiv:2601.00675*, 2026.

X. Li, K. Hsu, J. Gu, et al. Evaluating real-world robot manipulation policies in simulation. In *Conference on Robot Learning (CoRL)*, volume 270 of *PMLR*, pp. 3705–3728, 2024.

Y. Li, Y. Zhu, J. Wen, et al. WorldEval: World model as real-world robot policies evaluator. *arXiv preprint arXiv:2505.19017*, 2025.

Y. Li, J. Wen, Z. Zhou, et al. Scaling real-world robot policy evaluation via discrete diffusion world model. In *International Conference on Machine Learning (ICML)*, 2026. System name: dWorldEval.

C. Ma, T. Su, J. Zhu, et al. PiL-World: A chunk-wise world model for VLA policy-in-the-loop evaluation. *arXiv preprint arXiv:2606.05773*, 2026.

F. Mentzer, D. Minnen, E. Agustsson, et al. Finite scalar quantization: VQ-VAE made simple. In *International Conference on Learning Representations (ICLR)*, 2024.

E. Miller. Adding error bars to evals: A statistical approach to language model evaluations. *arXiv preprint arXiv:2411.00640*, 2024.

NVIDIA. Cosmos-Reason2: Physical AI common sense and embodied reasoning models. Model card, `https://huggingface.co/nvidia/Cosmos-Reason2-8B`, 2025. Released 2025-12-19; revised 2026-03-10.

A. Panickssery, S. R. Bowman, and S. Feng. LLM evaluators recognize and favor their own generations. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

K. Pertsch, K. Stachowicz, B. Ichter, et al. FAST: Efficient action tokenization for vision-language-action models. In *Robotics: Science and Systems (RSS)*, 2025.

J. Quevedo, A. K. Sharma, Y. Sun, et al. WorldGym: World model as an environment for policy evaluation. In *International Conference on Learning Representations (ICLR)*, 2026.

J. Rocamonde, V. Montesinos, E. Nava, et al. Vision-language models are zero-shot reward models for reinforcement learning. In *International Conference on Learning Representations (ICLR)*, 2024.

P. Selvaraj, L. Uttini, and V. Kuosmanen. ArmnetBench v0.1: Parallel real-world evaluation of manipulation policies on a low-cost arm farm. *arXiv preprint arXiv:2607.24481*, 2026.

S. A. Sontakke, J. Zhang, S. M. R. Arnold, et al. RoboCLIP: One demonstration is enough to learn robot policies. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

R. Tian, Y. Wu, and A. Bajcsy. Position: Good embodied reward models need bad behavior data. In *International Conference on Machine Learning (ICML) Position Paper Track*, 2026. Spotlight.

M. Tschannen, A. Gritsenko, X. Wang, et al. SigLIP 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features. *arXiv preprint arXiv:2502.14786*, 2025.

W.-C. Tseng, G. Hussein, Y. Dong, et al. SC3-Eval: Evaluating robot foundation models via self-consistent video generation. *arXiv preprint arXiv:2606.18610*, 2026.

P. Wang, L. Li, L. Chen, et al. Large language models are not fair evaluators. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2024.

S. Wang, H. Xu, Y. Hu, et al. A practical recipe towards improving sim-and-real correlation for VLA evaluation. *arXiv preprint arXiv:2606.10366*, 2026a.

Y. Wang, F. Ruan, R. Chen, et al. R2S-Eval: Robot evaluation with real-to-sim calibration via vision-language models. *arXiv preprint arXiv:2609.03276*, 2026b.

Y. Wang, Y. Zhan, Z. Feng, et al. TrustRoboReward: Preference-ordered isotonic score editing for multi-paradigm robot reward models. *arXiv preprint arXiv:2608.08491*, 2026c.

J. X. Yang, R. Dagli, A. Zook, et al. RoboLab: A high-fidelity simulation benchmark for analysis of task generalist policies. In *Robotics: Science and Systems (RSS)*, 2026.

J. Yuan, H. Li, X. Ding, et al. Understanding and mitigating numerical sources of nondeterminism in LLM inference. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

L. Zheng, W.-L. Chiang, Y. Sheng, et al. Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena. In *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track*, 2023.

Z. Zhou, P. Atreya, Y. L. Tan, et al. AutoEval: Autonomous evaluation of generalist robot manipulation policies in the real world. In *Conference on Robot Learning (CoRL)*, volume 305 of *PMLR*, pp. 1997–2017, 2025.

J. Zhu, W. Wang, Z. Chen, et al. InternVL3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*, 2025.

Appendix

The appendix gives the evidence behind each claim in the main text. Appendix A describes the audit setup and Appendix B the reference ranking and its population targets. Appendices C–D report selection and decisiveness (how often an evaluator avoids ties) for every evaluator and target (Section 4.1 and Section 4.2). Appendix E varies the readout and the aggregation, Appendix F the implementation, and Appendix G the set of policies. Appendix H covers the controls, Appendix I the published correlations, and Appendix J the registered hypotheses and their outcomes. Appendices K and L give extended related work and the released code and data.

A AUDIT SETUP

Figure 4 shows how the reference and each evaluator's ranking are built from the same sessions.

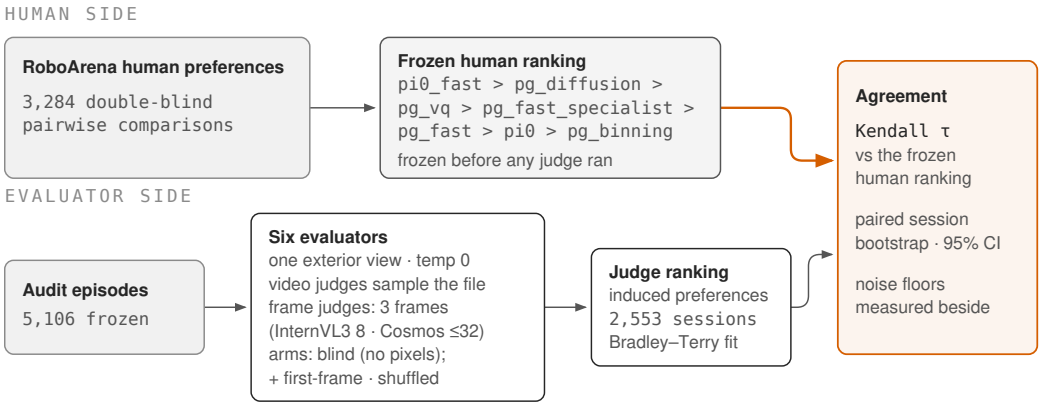

Figure 4: The registered campaign pipeline. RoboArena's double-blind comparisons fix a Bradley–Terry reference ranking over the seven policies with at least 600 episodes, frozen before any evaluator ran. Each evaluator scores the 5,106 audit episodes. Within each of the 2,553 sessions, its higher-scored episode is its preference (equal scores tie), and the same Bradley–Terry code yields its ranking. Reference and evaluator are refit on the same resampled sessions in every bootstrap draw. The decision-level measures (winner agreement, regret, frozen choice, held-out selection, reweighted targets) and the two added configurations are computed from the same scores.

Evaluator configurations. The evaluator's ranking uses the same Bradley–Terry code as the reference, and the two pipelines agree to 5×10^{-7}. Every evaluator sees one exterior camera, chosen in the fixed priority right, then left, then wrist. RoboReward-4B and -8B, Qwen2.5-VL-7B and -72B, and Cosmos-Reason2-8B read the video through their own processors. Cosmos is capped at 32 frames to fit in memory, which departs from its model card (5,043 of 5,106 episodes decode exactly 32 frames, and every exception is a clip too short to reach the cap). InternVL3-8B receives 8 frames and SigLIP2-B/16 receives 3 frames sampled by our code. The VLM judges and the reward models answer on the 1–5 progress scale of RoboReward's prompt. SigLIP2 scores the mean cosine between the instruction and each frame. All registered evaluators decode greedily (always taking the most likely token). Prompts, frame plans, and checkpoint revisions are in the released repository.

Parsing. A reply is scored if it contains an in-range answer. Cosmos declares an answer tag (a marked field that holds its answer) and sometimes writes the number without the registered prefix ("ANSWER:"). Accepting a bare in-range number inside its declared tag recovers 635 rows and lowers its parse failure from 12.6% to 0.16%. No other evaluator declares a tag, so the rule changes no other row. Under the stricter rule Cosmos exceeds the 5% compliance threshold and would be excluded. Its pick is unchanged (`pg_diffusion`, the runner-up), its τ becomes $+0.714$, and its winner agreement becomes 0.181, 0.044, and 0.051 under session, rater, and site resampling (0.233, 0.110, 0.112 under the widened rule). No conclusion depends on the rule (Table 5).

The two added configurations. Qwen2.5-VL-72B runs the registered Qwen protocol unchanged at 72B scale. LLM-as-a-Verifier (Kwok et al., 2026) reads a continuous progress score as the expectation

Table 5: Cosmos-Reason2 under the strict and the widened parse rule.

parse rule	sessions	τ	95% CI	parse failure
strict (prefix required)	1,964	$+0.714$	$[+0.238, +0.905]$	12.59%
widened (tag content)	2,545	$+0.619$	$[+0.238, +0.905]$	0.16%

over the answer tokens' probabilities. We run it on Qwen3.5-9B with temperature 1.0, granularity 20 (the number of answer letters), and four repeats, as in the library's example. The original paper's robotics result uses Qwen3.6-35B with eight repeats. The vLLM inference server cannot run on our hardware, so we serve the model through the `transformers` library. The verifier library's prompt, sampling request, and readout are unchanged. Because it samples at temperature 1.0, an independent full rerun reverses 9–10% of its induced preferences. The main tables report the original run, and Appendix F reports two further complete runs.

Pinned environment. All runs use torch 2.11.0, torchvision 0.26.0, and transformers 4.57.1 with the PyAV and decord video decoders. Decoder pairs agree to a maximum absolute pixel difference of 0. The pin affects cost as well as numerics. Under the older torch, a slow 3-D convolution kernel raises the campaign cost to 112 L40S-hours (NVIDIA GPU-hours) instead of 23.7.

Compute. Table 6 lists the GPU time of each scoring run. Analyses of stored scores run on CPU. The noise-floor shards and the implementation reruns of Appendix F are not itemized.

Table 6: GPU time of the scoring runs. The 72B model needs two 80 GB or larger cards. Everything else ran on one 48 GB L40S per process.

run	episodes	GPU time
six registered evaluators, audit set	5,106 each	23.7 L40S-hours (Cosmos-Reason2 8.8)
Qwen2.5-VL-72B, audit set	5,106	2.4 hours on 2 H100
LLM-as-a-Verifier, audit set	5,106	8.3 L40S-hours
LLM-as-a-Verifier, two further runs	5,106 each	about 9 L40S-hours each
expected-value readouts, five judges	5,106 each	13.0 L40S-hours (Cosmos-Reason2 9.6)
six registered evaluators, extension	1,462 each	3.6 L40S-hours
LLM-as-a-Verifier, extension	1,462	2.5 L40S-hours
Qwen2.5-VL-72B, extension	1,462	0.9 hours on 2 H200

B THE REFERENCE RANKING AND ITS POPULATION

The seven policies. All seven reference policies are vision-language-action models fine-tuned on DROID. Two start from the robot model π_0 (Black et al., 2024). Of these, `pi0` decodes actions by flow matching (a continuous generative method) and `pi0_fast` by FAST tokens (compressed discrete action codes) (Pertsch et al., 2025). The other five start from the VLM PaliGemma (Beyer et al., 2024), and they differ only in how they represent actions. `pg_diffusion` uses flow matching, like `pi0`. Both `pg_fast_spec` and `pg_fast` use FAST tokens, the first from a tokenizer trained on DROID and the second from the universal FAST+ tokenizer. `pg_vq` uses finite scalar quantization, which maps each action dimension to a fixed set of discrete levels (Mentzer et al., 2024). `pg_binning` splits each dimension into bins, as in RT-2 and OpenVLA (Brohan et al., 2023; Kim et al., 2024).

Table 7 gives the reference ranking with its intervals and the episodes behind each policy.

Across session draws, rank 1 is displaced in 6.0% of draws, ranks 2 and 3 exchange with probability 0.376, ranks 3 and 4 with 0.436, and ranks 4 and 5 with 0.224. Rank 7 is never displaced. Five of the 21 pairs have a bootstrap ordering probability (the share of draws that rank one above the other) between 0.2 and 0.8, so the reference cannot order them.

Fit of the reference model. The Bradley–Terry model fits the 21 pairwise win shares well: the sum of squared Pearson residuals is 12.7, and a parametric bootstrap from the fitted model exceeds it in 40% of 2,000 simulated datasets. The largest residual ($|1.73|$, `pi0_fast` over `pg_fast_spec`) is exceeded in 35%. Each pair has 68 to 228 sessions from 7 to 11 sites. Dropping ties instead

Table 7: The reference ranking, fit on the 2,553 sessions that compare two of the seven policies. Bradley–Terry log-strength θ with a 95% percentile interval over 10,000 session bootstrap draws. Ties count as half a win to each side, and mean log-strength is fixed at zero. *Episodes in dump*: the policy's episodes in all 3,284 sessions, which set eligibility. A group of 599 sessions records up to seven policies but compares only two, so 3,021 of these 8,815 episodes are never compared. These 599 sessions come from one early two-week phase of RoboArena. The preference always compares slots A and B (the two policy positions of a session). The policy in slot A is spread nearly evenly over the seven (66 to 99 sessions each), and all 21 pairs occur 20 to 40 times. This pattern fits slots assigned before the rollouts. *Audit comparisons*: its appearances in the 2,553 reference sessions ($2 \times 2{,}553 = 5{,}106$ in total).

rank	policy	episodes in dump	audit comparisons	θ	95% CI
1	`pi0_fast_droid`	1,168	642	$+0.648$	$[+0.514, +0.790]$
2	`paligemma_diffusion_droid`	1,171	654	$+0.472$	$[+0.345, +0.606]$
3	`paligemma_vq_droid`	1,198	660	$+0.443$	$[+0.309, +0.581]$
4	`paligemma_fast_specialist_droid`	1,392	847	$+0.427$	$[+0.312, +0.546]$
5	`paligemma_fast_droid`	1,387	851	$+0.365$	$[+0.252, +0.486]$
6	`pi0_droid`	1,431	898	$+0.113$	$[-0.005, +0.231]$
7	`paligemma_binning_droid`	1,068	554	-2.469	$[-2.785, -2.216]$

of counting them as half wins leaves the reference winner and its stability (0.935) unchanged, and changes no evaluator's pick.

Population targets. The dump records who rated each session and where. Table 8 lists the candidate clustering units (groupings of sessions whose outcomes may be correlated). Resampling sessions, tasks, rater-days, or days leaves interval widths unchanged. Dependence appears only at the rater and site level, where one cluster holds 53.7% of sessions. That cluster is one rater at one site, so "remove the dominant rater" and "remove the dominant site" are the same analysis. Table 9 states the question each target answers, and Table 10 shows the reference under each.

Table 8: Candidate clustering units in the RoboArena dump.

unit	clusters	largest	share in largest	median size	singletons
session	2,553	1	0.0%	1	2,553
task instruction	1,927	28	1.1%	1	1,649
rater × calendar day	323	50	2.0%	6	57
calendar day	153	139	5.4%	13	4
rater	44	1,370	53.7%	9	12
site	12	1,370	53.7%	95	1

Table 9: The question each target answers.

target	question
session resampling	would a fresh sample of sessions from this pool, with these raters, sites, and tasks, select the same policy?
rater resampling	would a fresh set of raters from the same pool, each bringing all their sessions, select it?
site resampling	would a fresh set of evaluation sites select it? With 12 sites, one holding over half the data, this is the coarsest unit the data support.
equal weight per site / rater	what would the pool select if every site (rater) counted equally, however many sessions it contributed?
dominant contributor removed	does the selection survive without the one rater and site that supply 53.7% of sessions? This removes that site's tasks too.

On the matched cohort, even under session resampling, the first two policies are separated by a fitted win probability of only 0.545 $[0.497, 0.594]$.

Table 10: The reference under each target. P(#1 beats #2) is the fitted probability that `pi0_fast` beats `pg_diffusion`, the full-data first and second, under that target.

target	sessions	winner	P(winner holds)	P(#1 beats #2) [95% CI]
session	2,553	`pi0_fast`	0.940	0.544 [0.495, 0.591]
rater	2,553	`pi0_fast`	0.686	0.544 [0.457, 0.590]
site	2,553	`pi0_fast`	0.682	0.544 [0.465, 0.589]
equal weight per site	2,553	`pg_fast_spec`	0.453	0.516 [0.446, 0.593]
equal weight per rater	2,553	`pg_vq`	0.523	0.606 [0.472, 0.729]
without site B	2,159	`pi0_fast`	0.977	0.560 [0.507, 0.612]
without site C	2,235	`pi0_fast`	0.889	0.534 [0.483, 0.585]
w/o dominant contributor	**1,183**	**`pg_vq`**	**0.512**	0.521 [0.455, 0.587]

Leaving out any one of the other eleven sites changes no evaluator's conclusion. Only leaving out the dominant contributor does (Table 11).

Table 11: Leave-one-site-out: range of each evaluator's τ over the 12 fits, and how many of the 12 intervals include zero.

evaluator	τ range, 12 fits	fits incl. zero	which site
RoboReward-4B	$+0.429$ to $+0.905$	0	—
RoboReward-4B clean	$+0.333$ to $+0.905$	1	dominant
RoboReward-8B	$+0.429$ to $+0.810$	0	—
RoboReward-8B clean	$+0.143$ to $+0.810$	1	dominant
Cosmos-Reason2-8B	$+0.429$ to $+0.714$	1	dominant
Qwen2.5-VL-7B	$+0.429$ to $+0.905$	0	—
InternVL3-8B	$+0.619$ to $+0.810$	0	—
SigLIP2-B/16	-0.810 to -0.333	1	dominant

C SELECTION UNDER EVERY TARGET

Weighted targets. For "sites equal" each session is weighted by $1/n_s$, where n_s is the number of sessions from its site, rescaled so the weights sum to the number of sessions. The weights enter the win counts of both the reference and the evaluator fit, and sites are resampled as clusters. "Raters equal" does the same with raters. With all weights equal to one the weighted fit reproduces the unweighted one exactly.

Regret distribution. Table 12 gives the full regret distribution under the session unit. Its last column is the probability, under the reference's own Bradley–Terry model, that the reference winner beats the evaluator's pick: for every evaluator with positive τ it averages between 0.506 and 0.540, close to a coin flip.

Table 12: Share of draws with regret at most δ (reference log-strength), session unit, matched cohort, and the mean probability that the reference winner beats the evaluator's pick.

evaluator	$\delta = 0.05$	0.10	0.20	0.30	0.50	P(winner beats pick)
RoboReward-4B	0.372	0.448	0.685	0.899	0.999	0.533
RoboReward-8B	0.474	0.563	0.801	0.957	1.000	0.524
Cosmos-Reason2-8B	0.214	0.309	0.615	0.885	0.999	0.540
Qwen2.5-VL-7B	0.359	0.456	0.739	0.937	1.000	0.530
InternVL3-8B	0.224	0.323	0.630	0.892	1.000	0.539
SigLIP2-B/16	0.000	0.000	0.000	0.000	0.000	0.957
Qwen2.5-VL-72B	0.627	0.702	0.872	0.975	1.000	0.516
LLM-as-a-Verifier	0.840	0.884	0.960	0.994	1.000	0.506
human progress score	0.996	1.000	1.000	1.000	1.000	0.500

SigLIP2 under every target. SigLIP2's τ interval includes zero when sites are resampled ($[-0.810, +0.048]$). Its regret, however, stays between 1.7 and 3.1 under five of the six resampling units and targets, and it is 1.1 when raters count equally.

Held-out selection. Table 14 separates the data a selector sees from the data that scores it. In each of 2,000 random splits of the matched cohort, stratified by policy pair, the reference is fit on one half (1,249 sessions) and each selector picks a policy from m sessions of the other half. Held-out regret is the reference log-strength the pick gives up. The held-out reference winner is `pi0_fast` in 0.950 of splits. Two baselines anchor the scale: a uniformly random policy, and a fixed choice of `pi0`, the first released generalist policy in the set (RoboArena has no predeclared incumbent). Even the raters' own preferences on all 1,249 selection sessions find the held-out winner in only 0.887 of splits, so part of the remaining regret is disagreement between the two halves of the reference.

Because the splits overlap, intervals come from an outer bootstrap. It draws 1,000 resamples of the sessions, stratified by policy pair, runs 100 fresh splits on each, and takes the 95% percentile interval of the mean held-out regret over the resamples. Differences are paired within each resample. Over all seven policies every evaluator except SigLIP2 beats a random pick by 0.47 to 0.61, with every interval excluding zero, but a random pick sometimes lands on `pg_binning`. Table 13 repeats the analysis on the 1,954 sessions that compare two of the six competitive policies. Among these, the random pick gives up 0.252 $[0.168, 0.385]$. We use its exact expectation, the reference winner's log-strength minus the mean log-strength, so the baseline adds no sampling noise. There the gain over a random pick is 0.06 to 0.11 with 200 selection sessions, and every interval excludes zero. With the full half (977 sessions), the gain is clear only for LLM-as-a-Verifier and Qwen2.5-VL-72B. No evaluator's gap to the raters' own preferences, and no gap between LLM-as-a-Verifier and another evaluator except SigLIP2, is established.

Table 13: Held-out selection among the six competitive policies: mean held-out regret and gain over a random pick (random minus selector), with 95% outer-bootstrap intervals (1,000 resamples), at 200 and at all 977 selection sessions. The random pick's regret is its exact expectation, 0.252 $[0.168, 0.385]$.

	held-out regret		gain over random	
selector	$m = 200$	$m = 977$	$m = 200$	$m = 977$
RoboReward-4B	0.185 $[0.104, 0.289]$	0.172 $[0.028, 0.325]$	0.067 $[0.009, 0.169]$	0.080 $[-0.032, 0.305]$
RoboReward-8B	0.176 $[0.103, 0.284]$	0.136 $[0.016, 0.291]$	0.076 $[0.012, 0.189]$	0.116 $[-0.003, 0.324]$
Cosmos-Reason2-8B	0.197 $[0.102, 0.328]$	0.184 $[0.055, 0.345]$	0.055 $[0.001, 0.141]$	0.068 $[-0.023, 0.212]$
Qwen2.5-VL-7B	0.172 $[0.097, 0.297]$	0.144 $[0.032, 0.297]$	0.080 $[0.014, 0.165]$	0.108 $[-0.003, 0.276]$
InternVL3-8B	0.185 $[0.103, 0.316]$	0.173 $[0.052, 0.338]$	0.067 $[0.010, 0.152]$	0.079 $[-0.018, 0.241]$
SigLIP2-B/16	0.322 $[0.195, 0.495]$	0.444 $[0.185, 0.652]$	-0.070 $[-0.172, 0.026]$	-0.192 $[-0.338, 0.042]$
Qwen2.5-VL-72B	0.146 $[0.087, 0.254]$	0.092 $[0.009, 0.261]$	0.106 $[0.037, 0.205]$	0.160 $[0.012, 0.336]$
LLM-as-a-Verifier	0.138 $[0.075, 0.250]$	0.073 $[0.004, 0.235]$	0.114 $[0.041, 0.222]$	0.179 $[0.040, 0.340]$
human progress score	0.112 $[0.073, 0.145]$	0.012 $[0.000, 0.116]$	0.140 $[0.056, 0.292]$	0.240 $[0.061, 0.385]$
human preferences	0.124 $[0.085, 0.151]$	0.026 $[0.000, 0.123]$	0.128 $[0.053, 0.284]$	0.226 $[0.051, 0.385]$

The decision versus the procedure. Winner agreement refits the evaluator in every draw, so it measures the stability of the selection procedure. Table 15 instead freezes the one pick each evaluator makes from the full data and measures its reference regret under resampling. It also reports *fixed-selector* targets, in which only the reference is reweighted (sites or raters counted equally, or the dominant contributor removed) while the evaluator keeps its unweighted selection rule. Under

Table 14: Held-out selection on the matched cohort: mean regret of the pick against a reference fit on disjoint sessions, by the number m of selection sessions, and how often the pick is the held-out winner at the full budget. 2,000 splits.

selector	mean held-out regret				picks held-out winner, $m = 1{,}249$
	$m = 50$	$m = 200$	$m = 800$	$m = 1{,}249$	
RoboReward-4B	0.200	0.187	0.182	0.185	0.296
RoboReward-8B	0.194	0.174	0.159	0.155	0.361
Cosmos-Reason2-8B	0.210	0.193	0.180	0.182	0.148
Qwen2.5-VL-7B	0.203	0.170	0.140	0.144	0.279
InternVL3-8B	0.210	0.190	0.179	0.179	0.162
SigLIP2-B/16	1.629	2.651	3.123	3.123	0.000
Qwen2.5-VL-72B	0.186	0.155	0.117	0.103	0.527
LLM-as-a-Verifier	0.184	0.138	0.077	0.050	0.757
human progress score	0.169	0.124	0.045	0.014	0.923
human preferences	0.167	0.131	0.063	0.029	0.887
random pick (expectation)	0.658	0.658	0.658	0.658	0.143
fixed choice of `pi0`	0.557	0.557	0.557	0.557	0.000

these, winner agreement falls below its session-unit value for every evaluator and for the human progress score alike, to 0.09–0.23 for the three evaluators that pick `pi0_fast` and to 0.17–0.28 for the human progress score, because the reweighted reference names a different winner. Against the reweighted references, the fixed full-data pick `pi0_fast` gives up 0.13 to 0.28 of reference strength, the runner-up pick 0.24 to 0.62, and SigLIP2's pick 1.8 to 2.5 (Table 16).

Table 15: Frozen-choice regret (session unit) and winner agreement under fixed-selector targets (reference reweighted, evaluator unchanged), matched cohort.

evaluator	frozen pick	frozen pick, session unit		fixed-selector winner agreement		
		P(reference winner = pick)	mean regret	sites equal	raters equal	w/o dominant
RoboReward-4B	`pg_diffusion`	0.033	0.182	0.228	0.153	0.149
RoboReward-8B	`pi0_fast`	0.945	0.002	0.170	0.138	0.165
Cosmos-Reason2-8B	`pg_diffusion`	0.033	0.182	0.063	0.034	0.061
Qwen2.5-VL-7B	`pg_diffusion`	0.033	0.182	0.058	0.078	0.081
InternVL3-8B	`pg_diffusion`	0.033	0.182	0.101	0.138	0.096
SigLIP2-B/16	`pg_binning`	0.000	3.125	0.001	0.001	0.000
Qwen2.5-VL-72B	`pi0_fast`	0.945	0.002	0.089	0.183	0.124
LLM-as-a-Verifier	`pi0_fast`	0.945	0.002	0.101	0.229	0.149
human progress score	`pi0_fast`	0.945	0.002	0.197	0.278	0.171

Table 16: Regret of each evaluator's frozen full-data pick against the reweighted references, matched cohort. Evaluators with the same pick share a row.

frozen pick	evaluators	sites equal	raters equal	w/o dominant
`pi0_fast`	RoboReward-8B, Qwen2.5-VL-72B, LLM-as-a-Verifier, human progress score	0.283	0.219	0.135
`pg_diffusion`	RoboReward-4B, Cosmos-Reason2-8B, Qwen2.5-VL-7B, InternVL3-8B	0.352	0.617	0.235
`pg_binning`	SigLIP2-B/16	1.763	2.469	2.464

Intervals for differences in winner agreement. Winner agreement is itself a probability over bootstrap draws, so a difference between two evaluators needs a second level of resampling. We draw 200 outer resamples of the sessions and, on each, 500 paired inner draws, and take the 95% percentile interval of the difference over the outer draws (Table 17). The evaluator contrasts use 400

outer by 1,000 inner draws. For the two that also have a 200 by 500 version, no endpoint moves by more than 0.053. The interval for each evaluator's own winner agreement is also wide under this procedure (Table 2), for example 0.419 $[0.07, 0.91]$ for RoboReward-8B and 0.795 $[0.20, 0.98]$ for LLM-as-a-Verifier. Every interval for a tested difference includes zero. Given the small gaps between the leading policies and this many sessions, the differences in winner agreement reported in Section 4.1 are observed, not established.

A simulation checks this interval near ties. It keeps the audit's 2,553 sessions and their policy pairs. The true strengths are the frozen reference, with the top policy moved to the second policy's strength plus a gap of 0, 0.05, the observed 0.176, or 0.4. Each session draws one latent (unobserved true) outcome from the Bradley–Terry model at the true strengths. The simulated human reports it, or a tie at the observed human tie rate (0.145). Each of two synthetic evaluators ties at its own observed rate and otherwise reports the latent outcome with probability q and its reverse otherwise. We set q so that its half-credit accuracy (a tie against a decided session earns half credit) against the human equals the observed value (RoboReward-8B: accuracy 0.750, tie rate 0.404, $q = 0.934$; LLM-as-a-Verifier: 0.716, 0, 0.753). Sharing the latent outcome makes both evaluators correlate with the human as paired data do. The true winner agreement of each evaluator, and so the true difference, is the average over 20,000 fresh simulated audits. Each of 200 datasets per gap then gets the full 400 by 1,000 double bootstrap. The nominal 95% interval for their difference covers the true difference in 0.96 to 1.00 of datasets (0.99 at the observed gap), so it is conservative. Intervals for a single evaluator's winner agreement under-cover (0.84 to 0.88) only when the true value is near 1. The simulation resamples sessions and does not check the rater and site units. These have 44 and 12 clusters, with one rater and site holding 53.7% of sessions, so we read those columns as sensitivity analyses, not as calibrated intervals.

Matching accuracy and tie rate does not fix how an evaluator errs on the leading pair, so a second simulation targets it (at the observed gap, 200 datasets, 200 outer by 500 inner draws). A twin of each synthetic evaluator keeps its accuracy and tie rate but aims its errors at the top: it reverses the comparisons of the two strongest policies with probability 0.7, or turns a loss of the runner-up into a win with probability 0.12 or 0.28 in every session involving it. Its true winner agreement falls by 0.11 to 0.18. The interval for the difference still covers the truth in 0.980 to 0.995 of datasets, but it excludes zero in at most 1% of them. At this gap and sample size, the interval is valid but almost powerless against such a loss. The estimated winner agreement also falls below the true value (for example 0.59 against 0.76 for the RoboReward-8B-like evaluator), so observed winner agreement understates the stability of a good evaluator.

Table 17: Double-bootstrap 95% intervals for differences in winner agreement, session unit. The readout and aggregation rows use each evaluator's own sessions.

difference	95% interval
LLM-as-a-Verifier − InternVL3-8B (matched cohort)	$[-0.111, +0.864]$
LLM-as-a-Verifier − RoboReward-8B (matched cohort)	$[-0.284, +0.705]$
LLM-as-a-Verifier − RoboReward-4B (matched cohort)	$[-0.197, +0.826]$
expected value − discrete, RoboReward-4B	$[-0.244, +0.490]$
expected value − discrete, RoboReward-8B	$[-0.304, +0.514]$
expected value − discrete, Qwen2.5-VL-7B	$[-0.206, +0.532]$
expected value − discrete, InternVL3-8B	$[-0.338, +0.361]$
expected value − discrete, Cosmos-Reason2-8B	$[-0.402, +0.252]$
mean score − paired comparison, Qwen2.5-VL-72B	$[-0.150, +0.581]$
mean score − paired comparison, LLM-as-a-Verifier	$[-0.293, +0.550]$

Each evaluator's own sessions. Table 2 uses the 2,498 sessions every evaluator scored. Cosmos scored 2,545 sessions, LLM-as-a-Verifier 2,506, and the others all 2,553. Table 18 repeats the audit on each evaluator's own sessions, where the reference is refit on the same sessions. The conclusions of Section 4 hold. One pick changes. On its full 2,553 sessions, RoboReward-4B selects the reference winner, and its τ rises from $+0.619$ to $+0.810$. The change in its selection comes from 2% of the sessions. Those 55 sessions are the ones LLM-as-a-Verifier (47) or Cosmos (8) could not score. They span 17 of the 21 policy pairs, 10 sites, and 15 raters, and the dominant rater supplies 45% of them, against 54% overall.

Table 18: The audit of Table 2 on each evaluator's own sessions. Bold: the highest observed winner agreement among automatic evaluators under each target.

evaluator	τ 95% CI	pick rank	resampling unit: session	rater	site	reference target: sites equal	raters equal	w/o dominant	regret res./frozen
reference winner holds in			0.940	0.686	0.682	0.453	0.523	0.512	
RoboReward-4B	$+0.810$ $[+0.238, +0.905]$	1	0.398	0.519	0.567	0.359	0.269	0.279	0.109 0.002
RoboReward-8B	$+0.714$ $[+0.333, +0.905]$	1	0.422	0.485	0.524	**0.485**	0.269	0.298	0.094 0.002
Cosmos-Reason2-8B	$+0.619$ $[+0.238, +0.905]$	2	0.233	0.110	0.112	0.096	0.246	0.109	0.133 0.168
Qwen2.5-VL-7B	$+0.714$ $[+0.333, +0.905]$	2	0.392	0.247	0.262	0.230	0.063	0.205	0.102 0.179
InternVL3-8B	$+0.714$ $[+0.238, +0.905]$	2	0.205	0.231	0.271	0.201	0.236	**0.414**	0.148 0.179
SigLIP2-B/16	-0.714 $[-0.905, -0.238]$	7	0.000	0.002	0.001	0.005	0.090	0.001	3.134 3.134
Qwen2.5-VL-72B	$+0.714$ $[+0.429, +0.905]$	1	0.591	0.529	0.519	0.115	**0.309**	0.184	0.063 0.002
LLM-as-a-Verifier	$+0.810$ $[+0.429, +1.000]$	1	**0.827**	**0.661**	**0.614**	0.387	0.250	0.258	0.023 0.001
human progress score	$+0.905$ $[+0.714, +1.000]$	1	0.954	0.850	0.821	0.691	0.706	0.616	0.001 0.002

Rank statistics in full. Table 19 gives Spearman's ρ (the Pearson correlation of the ranks) beside τ, with the number of sessions and induced ties. An exact permutation test of each τ against a uniformly random ranking of seven policies gives one-sided p of 0.0054 ($\tau = 0.810$), 0.0151 (0.714), and 0.0345 (0.619). The smallest τ significant at 0.05 is 0.619. Bias-corrected and accelerated bootstrap intervals are unreliable for a statistic that takes 22 values and often does not change when one session is removed, so we report percentile intervals throughout.

Table 19: Rank statistics, session unit, 10,000-draw paired session bootstrap.

evaluator	τ	95% CI	ρ	sessions	ties
RoboReward-4B	$+0.810$	$[+0.238, +0.905]$	$+0.893$	2,553	1,043
clean	$+0.714$	$[+0.238, +0.905]$	$+0.786$	1,839	748
RoboReward-8B	$+0.714$	$[+0.333, +0.905]$	$+0.786$	2,553	1,031
clean	$+0.714$	$[+0.238, +0.905]$	$+0.786$	1,839	748
Cosmos-Reason2-8B	$+0.619$	$[+0.238, +0.905]$	$+0.821$	2,545	1,052
Qwen2.5-VL-7B	$+0.714$	$[+0.333, +0.905]$	$+0.857$	2,553	922
InternVL3-8B	$+0.714$	$[+0.238, +0.905]$	$+0.857$	2,553	1,434
SigLIP2-B/16	-0.714	$[-0.905, -0.238]$	-0.821	2,553	0
Qwen2.5-VL-72B	$+0.714$	$[+0.429, +0.905]$	$+0.857$	2,553	1,227

Benchmark error and ranking. Three audited evaluators appear on RoboRewardBench (RoboReward's leaderboard), whose mean absolute error (MAE) against human progress labels differs by a factor of 1.8 among them (0.665, 0.845, 1.172). Their τ cannot be separated (paired $\Delta\tau$ intervals all include zero, and the smallest detectable difference at 80% power is 0.46 to 0.56), so this audit cannot tell whether benchmark error predicts ranking fidelity.

D DECISIVENESS

Table 20 reports how often each evaluator commits to a preference and how accurate it is. Discrete evaluators tie often. InternVL3 ties on 56% of sessions and the 72B judge on 48%. The 72B judge answers 1 or 5 on 77% of episodes and uses the middle of the scale on 5%. SigLIP2 never ties and scores 0.432 in half-credit accuracy (0.420 where both commit), the pairwise counterpart of its

Table 20: Decisiveness. Pairwise accuracy scores a session 1 when evaluator and reference make the same call (including both tying), 0.5 when exactly one ties, and 0 for opposite calls. Brackets are 95% paired session-bootstrap intervals. For the two proportions, Wilson intervals agree with them to within 0.002. The reference is directional in 85.5% of sessions (1,101 A, 1,081 B, 371 ties). *Both commit*: neither evaluator nor reference ties. Baselines for the half-credit accuracy: always tie 0.573, coin flip 0.500.

evaluator	non-tie rate	accuracy (half credit)	sessions where both commit	accuracy where both commit
RoboReward-4B	0.591 $[0.572, 0.611]$	0.733 $[0.721, 0.746]$	1,380	0.844 $[0.824, 0.863]$
RoboReward-8B	0.596 $[0.577, 0.616]$	0.750 $[0.738, 0.763]$	1,396	0.870 $[0.853, 0.888]$
Cosmos-Reason2-8B	0.587 $[0.567, 0.606]$	0.667 $[0.653, 0.680]$	1,317	0.749 $[0.725, 0.772]$
Qwen2.5-VL-7B	0.639 $[0.620, 0.658]$	0.706 $[0.693, 0.719]$	1,452	0.795 $[0.775, 0.816]$
InternVL3-8B	0.438 $[0.419, 0.458]$	0.650 $[0.638, 0.663]$	995	0.761 $[0.734, 0.787]$
SigLIP2-B/16	1.000 $[1.000, 1.000]$	0.432 $[0.414, 0.450]$	2,182	0.420 $[0.400, 0.441]$
Qwen2.5-VL-72B	0.519 $[0.500, 0.539]$	0.711 $[0.699, 0.723]$	1,203	0.845 $[0.825, 0.866]$
LLM-as-a-Verifier	1.000	0.716	—	0.753

negative τ. High pairwise accuracy does not imply good selection. Over all the sessions each scored, RoboReward-8B's half-credit accuracy exceeds LLM-as-a-Verifier's (0.750 against 0.716), yet it recovers the reference winner far less often.

Confident ordering of pairs the reference cannot order. Every evaluator orders at least one of the five ambiguous reference pairs with at least 95% bootstrap probability: RoboReward-4B 3 of 5, RoboReward-8B 2, Cosmos 2 of 4 (its scored sessions leave four ambiguous pairs), Qwen2.5-VL-7B 2, InternVL3 1, SigLIP2 2, Qwen2.5-VL-72B 1, and LLM-as-a-Verifier 1 of 4. This is a difference in decisiveness between evaluator and reference, and it does not show which ordering is correct.

E Readout and aggregation

An evaluator's selection depends on two choices made after the model runs: how an answer becomes a score (the readout) and how scores become a ranking (the aggregation). This section varies each with the model held fixed.

Expected-value readout. For each registered judge with a 1–5 answer we re-ran the campaign input (same checkpoint, prompt, frames, decoder, and kernel), fed the model its own answer prefix, and read the next-token distribution over the digits 1 to 5. The score is the expectation $\sum_k k\, p_k$ of the renormalised distribution, where p_k is the probability of digit k. For Cosmos-Reason2, the distribution is read at the answer digit after its reproduced chain-of-thought. The most likely digit equals the recorded campaign score on all 5,106 episodes for each of the four short-answer judges and on all 5,098 episodes the campaign parsed for Cosmos, so the two readouts differ only in how the same distribution is turned into a number. Table 21 compares them on the same resamples.

Under session resampling, the expected value raises winner agreement for three of the five judges (Qwen2.5-VL-7B from 0.392 to 0.633) and lowers it for two, while τ moves the other way for three. Under rater and site resampling the direction is the same: the expected value raises winner agreement for RoboReward-4B (0.519 to 0.621 by rater, 0.567 to 0.646 by site), RoboReward-8B (0.485 to 0.585, 0.524 to 0.600), and Qwen2.5-VL-7B (0.247 to 0.509, 0.262 to 0.496), and lowers it for InternVL3 and Cosmos. For Cosmos, the expected value raises τ to $+0.905$ under all three units while lowering winner agreement. A continuous readout changes selection and rank agreement by different amounts and often in different directions.

Inside LLM-as-a-Verifier. The same recorded run can be read two ways: the continuous expectation over answer-token probabilities, or the most likely answer letter. On the 2,419 sessions where both readings exist, the continuous reading gives $\tau = +0.810$ $[+0.524, +1.000]$ and winner agreement 0.878. The discrete reading gives $\tau = +0.524$ $[+0.238, +0.905]$, winner agreement 0.681, and ties on 69% of sessions. RoboReward-4B reaches 0.425 on the same episodes. The paired τ

Table 21: Discrete (most likely digit) and expected-value readouts of the same model outputs, session unit. Δ: expected value minus discrete, paired over the same draws. *Only one agrees*: draws in which exactly one readout picks the reference winner, counted for each readout (EV: expected value). *Tie resolution*: of the sessions the discrete readout ties but the raters decide, how many the expected value decides as the raters did. The expected value removes nearly all ties. The discrete tie rates are 41%, 40%, 36%, 56%, and 41% in row order. Bold: the readout with the higher winner agreement.

	τ		winner agreement			
judge	discrete	Δ [95% CI]	discrete	expected value	only one agrees (EV / discrete)	tie resolution
RoboReward-4B	$+0.810$	-0.190 $[-0.286, +0.286]$	0.398	**0.521**	1,997 / 772	540 / 802
RoboReward-8B	$+0.714$	-0.095 $[-0.286, +0.286]$	0.422	**0.602**	2,460 / 655	531 / 786
Qwen2.5-VL-7B	$+0.714$	$+0.095$ $[-0.286, +0.381]$	0.392	**0.633**	2,905 / 489	460 / 730
InternVL3-8B	$+0.714$	-0.190 $[-0.381, +0.286]$	**0.205**	0.145	652 / 1,253	774 / 1,187
Cosmos-Reason2-8B	$+0.619$	$+0.286$ $[-0.286, +0.381]$	**0.233**	0.165	515 / 1,190	502 / 859

difference, -0.286 $[-0.571, +0.286]$, includes zero, so readout resolution is a candidate explanation of this configuration's high winner agreement, not an established one.

Mean score instead of paired comparison. The registered aggregation compares the two episodes of each session, which holds the task fixed. Table 22 ranks policies by their mean episode score instead, on the same sessions and resamples. The mean score lowers τ for every evaluator with positive τ, including the human progress score ($+0.905$ to $+0.429$), except Qwen2.5-VL-72B, whose τ is unchanged. It ignores which policy each episode faced on which task. Under it, five of the nine rows of Table 22 no longer beat a random ranking by exact permutation test. The paired comparison gives the higher observed rank agreement for most configurations in this audit. Selection moves the other way for evaluators with many ties or extreme answers: winner agreement rises for Qwen2.5-VL-72B (0.591 to 0.889), LLM-as-a-Verifier (0.827 to 0.915), Cosmos (0.233 to 0.415), and Qwen2.5-VL-7B (0.392 to 0.401), but regret falls only for the first two (0.063 to 0.013 and 0.023 to 0.011). For Cosmos and Qwen2.5-VL-7B, regret rises slightly. By the decision criterion, mean score is the better rule for Qwen2.5-VL-72B and LLM-as-a-Verifier. But the double-bootstrap intervals of these changes include zero (Table 17), and mean scores are exposed to the task mix that pairing controls. Neither rule is established as better, which is why the report card states the rule.

Table 22: Induced-preference Bradley–Terry (registered) against mean score per policy, session unit, same sessions and resamples. Pick: reference rank of the evaluator's best policy.

		paired comparison (Bradley–Terry)				mean score			
evaluator	ties	τ	agree	pick	regret	τ	agree	pick	regret
RoboReward-4B	0.409	$+0.810$	0.398	1	0.109	$+0.429$	0.319	5	0.161
RoboReward-8B	0.404	$+0.714$	0.422	1	0.094	$+0.429$	0.288	5	0.169
Cosmos-Reason2-8B	0.413	$+0.619$	0.233	2	0.133	$+0.524$	0.415	1	0.136
Qwen2.5-VL-7B	0.361	$+0.714$	0.392	2	0.102	$+0.619$	0.401	2	0.109
InternVL3-8B	0.562	$+0.714$	0.205	2	0.148	$+0.333$	0.032	2	0.201
SigLIP2-B/16	0.000	-0.714	0.000	7	3.134	-0.810	0.000	7	3.048
Qwen2.5-VL-72B	0.481	$+0.714$	0.591	1	0.063	$+0.714$	0.889	1	0.013
LLM-as-a-Verifier	0.000	$+0.810$	0.827	1	0.023	$+0.619$	0.915	1	0.011
human progress score	0.195	$+0.905$	0.954	1	0.001	$+0.429$	0.864	1	0.018

Combining evaluators. A stacked re-scoring learns a weighted combination of the six registered evaluators' scores (13 features, pairwise logistic regression with a ridge penalty, which shrinks the weights toward zero) under leave-one-policy-out cross-validation. Only the six pairs that involve the held-out policy are unseen by the fit, so we score each fold on those pairs alone. Concordance with

the reference is $+0.762$ $[+0.381, +0.952]$, against $+0.810$ for RoboReward-4B (paired difference -0.048 $[-0.238, +0.429]$). The combination picks the reference winner in 2 of 7 folds. An equal-weight mean of the six z-scored evaluators, with nothing fitted, gives $\tau = +0.619$ and picks the runner-up. Combining these evaluators does not improve on the best single one at this sample size.

F IMPLEMENTATION SENSITIVITY

Noise floors. Table 23 re-runs a 64-episode shard (a fixed subset of episodes) of each evaluator under an equally valid configuration: the same checkpoint, prompt, decoder, and decoded pixels, with one axis changed. In exact arithmetic the two runs would agree, so any difference comes from the order of floating-point operations. A same-environment control across two nodes gives 64/64 byte-identical scores and text, so every change below is caused by the named axis, meaning the setting that was changed. A noise floor must name its axis because the axis sets its size. InternVL3 moves 1.6% under a torch-version change and 17.2% under an attention-kernel change. Cosmos, which writes a chain-of-thought before its answer, moves most. Of its 64 recorded scores, 21 change (32.8%). Of these, 13 only switch the answer format between "ANSWER: 5" and a bare "5", which the widened parse rule reads identically, and 8 (12.5%) change the verdict. On a larger 1,024-episode sample the verdict changes on 23.1% of episodes. A small numerical difference early in a long generation can change the rest of the reasoning, as reported for reasoning models in text (Yuan et al., 2025). In one episode (task: rotate a bottle by 90 degrees) the two chains agree on the first sentence and then describe different videos. Under one torch version: "the robot arm moved towards the bottle, adjusted its position, made contact, and then lifted it slightly off the couch while maintaining grip", score 2. Under the other: "the robot arm moves towards the bottle but doesn't make contact or manipulate it", score 1. The pixels are identical, so the divergence is in what the model reports perceiving.

Table 23: Noise floors: share of 64 episode scores that change. The bracket counts scores that stayed identical. Sdpa and eager are PyTorch's fused and plain attention implementations.

evaluator	torch 2.11 vs. 2.9.1	attention sdpa vs. eager
Cosmos-Reason2-8B	12.5% verdicts (32.8% incl. format)	not measured
Qwen2.5-VL-7B	4.7% (61/64)	not measured
InternVL3-8B	1.6% (63/64)	**17.2% (53/64)**
RoboReward-4B	0% (64/64, text too)	not measured
RoboReward-8B	0% (64/64, text too)	not measured
SigLIP2-B/16	39.1%, all by $\leq 2.5 \times 10^{-8}$; no preference flips	not measured

For InternVL3 we also separated checkpoint conversion from kernel. The converted checkpoint matches the published one tensor for tensor (all 637 tensors byte-identical by content hash, plus 48 fused attention tensors that appear as their three slices), and the attention kernel accounts for about three quarters of the difference between the two inference paths.

Measured reruns. Two evaluators were re-scored in full or in part under the second configuration. Table 24 separates the levels at which a change can appear. InternVL3's full rerun changes 14.6% of episode scores and 18.1% of session outcomes, but most of those are ties forming or breaking. Only 1.4% of sessions reverse, one adjacent pair of policies exchanges ranks, and the winner does not change. For Cosmos, a full rerun would cost about 69 L40S-hours. A 1,024-episode sample stratified by policy changes 23.1% of scores, but its campaign scores already give a different winner than the full set, so the sample cannot settle the winner question.

A sensitivity scenario. To ask how far the measured per-episode changes could move a leaderboard, we apply each evaluator's 64-episode transition matrix (how often each score changed to each other score) independently to every episode and refit (10,000 replicates). For InternVL3 on the attention axis the scenario gives a winner change in 0.281 of replicates. It assumes independent perturbations, treats a transition matrix estimated from 64 episodes as known, and holds unseen scores fixed. The data do not support these assumptions. It also overstated InternVL3's measured outcome: in the full rerun the winner did not change. The Cosmos scenario used recorded scores that include answer-format changes, so we do not report it.

Table 24: Measured reruns under an equally valid configuration, by the level at which the change appears.

level	InternVL3, attention kernel, full	Cosmos, torch version, sample
episode scores changed	747 / 5,105 (14.6%)	236 / 1,021 (23.1%)
session outcomes changed	463 / 2,552 (18.1%)	—
of which a tie formed or broke	428 (16.8%)	—
of which A and B reversed	35 (1.4%)	—
adjacent rank exchanges	1	—
$\Delta\tau$ [95% CI]	-0.095 $[-0.190, +0.381]$	-0.095 $[-0.476, +0.381]$
winner changed	no	not determinable

LLM-as-a-Verifier's own noise floor. On a 64-episode shard, rerunning with nothing changed flips 2 of 31 induced preferences (6.5%) and the attention-kernel swap flips 4 of 31 (12.9%). With eight repeats instead of four the replicate correlation rises from 0.954 to 0.977, but the flips remain (2 of 32, and 3 of 32). They sit in sessions whose two scores are close.

Three complete runs. We reran the configuration twice more on all 5,106 episodes, with identical inputs (prompt digests, decoded pixels, and frame indices match on every episode) and only the sampling randomness changed. Episode scores correlate at Pearson $r = 0.953$ to 0.959 between runs, and 9.3% to 10.2% of induced preferences reverse. The selection is stable (Table 25). Every run picks `pi0_fast`, the reference winner, under every resampling unit, and the original run has the lowest winner agreement of the three, so the value in the main tables is the most conservative.

Table 25: LLM-as-a-Verifier in three independent complete runs, own sessions.

unit	run	τ	pick	winner agreement	mean regret
session	original	$+0.810$	`pi0_fast`	0.827	0.023
	rerun 1	$+0.810$	`pi0_fast`	0.863	0.017
	rerun 2	$+0.905$	`pi0_fast`	0.916	0.009
rater	original	$+0.810$	`pi0_fast`	0.661	0.052
	rerun 1	$+0.810$	`pi0_fast`	0.759	0.031
	rerun 2	$+0.905$	`pi0_fast`	0.722	0.037
site	original	$+0.810$	`pi0_fast`	0.614	0.053
	rerun 1	$+0.810$	`pi0_fast`	0.717	0.036
	rerun 2	$+0.905$	`pi0_fast`	0.712	0.034

G MORE POLICIES: THE INCLUSION THRESHOLD

We scored both episodes of the 731 sessions outside the seven-policy reference with all eight evaluators, each under its original configuration, and refit reference and evaluators on the sessions that compare two policies meeting each rule. The seven-policy numbers reproduce the campaign exactly. With all fifteen policies the reference uses a small ridge penalty, because seven policies have too few comparisons for a maximum likelihood fit (343 of 10,000 draws separate: a policy wins or loses all its comparisons, so the fit has no finite solution). Its winner has only 58 episodes and holds in only 0.667 of draws, so that reference is too thin to audit against. Table 26 gives every evaluator under each rule.

Six policies: the clearly worst policy removed. The seven-policy reference places `pg_binning` far below the rest (log-strength -2.47 against $+0.11$ to $+0.65$). To see how much rank agreement comes from separating that one policy, we refit on the 1,999 sessions that compare two of the other six. The τ point estimates of the seven positive evaluators barely move ($+0.600$ to $+0.733$, against $+0.619$ to $+0.810$), but their intervals widen, and three now include zero (RoboReward-4B, InternVL3, and Cosmos). No pick changes, and winner agreement stays between 0.231 and 0.721. The agreement therefore holds among the six competitive policies, but part of its statistical certainty came from the worst one. SigLIP2 now picks `pi0`, the worst of the six, with a regret of 0.47.

Table 26: Evaluators under each inclusion rule, session unit.

rule	evaluator	sessions	τ [95% CI]	winner agreement	regret	pick
≥ 300	RoboReward-4B	3,087	$+0.857$ $[+0.286, +0.857]$	0.444	0.113	`pi05`
	RoboReward-8B	3,087	$+0.857$ $[+0.357, +0.857]$	0.531	0.084	`pi05`
	Cosmos-Reason2-8B	3,077	$+0.500$ $[+0.143, +0.786]$	0.098	0.223	`pg_diffusion`
	Qwen2.5-VL-7B	3,087	$+0.714$ $[+0.286, +0.857]$	0.413	0.134	`pg_diffusion`
	InternVL3-8B	3,087	$+0.714$ $[+0.357, +0.929]$	0.579	0.077	`pi05`
	Qwen2.5-VL-72B	3,087	$+0.714$ $[+0.429, +0.929]$	0.412	0.101	`pi0_fast`
	LLM-as-a-Verifier	3,022	$+0.786$ $[+0.357, +0.857]$	0.197	0.128	`pi0_fast`
	SigLIP2-B/16	3,087	-0.357 $[-0.500, +0.071]$	0.000	3.309	`pg_binning`
all	RoboReward-4B	3,284	$+0.276$ $[+0.124, +0.771]$	0.439	0.631	`dreaming_zebra`
	RoboReward-8B	3,284	$+0.543$ $[+0.238, +0.810]$	0.686	0.245	`dreaming_zebra`
	Cosmos-Reason2-8B	3,274	$+0.562$ $[+0.143, +0.752]$	0.567	0.452	`dreaming_zebra`
	Qwen2.5-VL-7B	3,284	$+0.714$ $[+0.181, +0.810]$	0.548	0.461	`dreaming_zebra`
	InternVL3-8B	3,284	$+0.257$ $[+0.067, +0.714]$	0.380	0.808	`floppy_bagel_90k`
	Qwen2.5-VL-72B	3,284	$+0.657$ $[+0.200, +0.790]$	0.665	0.323	`dreaming_zebra`
	LLM-as-a-Verifier	3,214	$+0.657$ $[+0.219, +0.790]$	0.568	0.413	`dreaming_zebra`
	SigLIP2-B/16	3,284	-0.219 $[-0.486, +0.143]$	0.054	2.005	`floppy_bagel_90k`

H CONTROLS

Contamination is label leakage. RoboReward's training set includes RoboArena episodes with their human progress scores mapped to its 1–5 scale (Section 4.1 of Lee et al., 2026). 1,322 of our 5,106 episodes are in that set, so on those episodes RoboReward was trained on the labels whose companion preferences form the reference. We report RoboReward on the 1,839 sessions with both episodes unseen. Those sessions are not a random subset. The excluded 714 sessions come almost entirely from the other raters (12 of them from the dominant contributor). The dominant contributor's share therefore rises from 53.7% to 73.8% in the clean cohort, and the clean reference depends even more on one rater. Clean rows are therefore compared with a different reference, and differences between clean and full rows are not a decontamination effect. Even the clean episodes share the rater pool whose rubric RoboReward learned. Table 27 gives every evaluator on these sessions.

The human progress score as an evaluator. RoboArena records a per-episode progress score beside each preference. Passing it through the same pipeline shows what the audit reports for a score that comes from the raters themselves. It recovers the reference ($\tau = +0.905$ $[+0.714, +1.000]$, winner agreement 0.954). The coarser binary success flag, which ties on 86% of sessions, still reaches $\tau = +0.619$. Score and preference come from the same rater in the same sitting and agree on 94.1% of sessions. The control shows that the aggregation can rebuild the reference from a consistent per-episode score. It cannot tell whether the evaluators err from low accuracy or because they judge a different criterion: the rater watched the robot with full information, while each evaluator sees one exterior camera. A human rating from the evaluators' input would separate the two, but we did not run it.

Table 27: All evaluators on the 1,839 sessions with both episodes outside RoboReward's training data. The reference is refit on the same sessions. Session unit unless stated.

evaluator	sessions	τ	pick (rank)	winner agreement	mean regret
RoboReward-4B	1,839	$+0.714$	1	0.682	0.086
RoboReward-8B	1,839	$+0.714$	1	0.692	0.081
Cosmos-Reason2-8B	1,834	$+0.619$	2	0.207	0.231
Qwen2.5-VL-7B	1,839	$+0.714$	1	0.534	0.127
InternVL3-8B	1,839	$+0.619$	2	0.241	0.230
SigLIP2-B/16	1,839	-0.524	7	0.000	3.727
Qwen2.5-VL-72B	1,839	$+0.714$	1	0.592	0.105
LLM-as-a-Verifier	1,811	$+0.810$	1	0.604	0.103
human progress score	1,839	$+0.810$	1	0.994	0.000

Blind arms cannot measure a language prior here. Each registered evaluator also ran with no pixels, to measure its language prior (a preference from the instruction text alone). Both episodes in a RoboArena session share one instruction, so the two blind prompts are byte-identical, every blind arm ties every session it scored, and τ is undefined by construction. The identical prompts also give a large determinism check. In the InternVL3 and Qwen2.5-VL-72B blind arms, greedy decoding returned identical text for both prompts of all 2,553 sessions. A duration-only control (longer episode preferred) is informative on only 488 sessions, because 2,063 of the 2,065 equal-duration pairs both hit the 399-second recording limit. It gives $\tau = +0.238$ $[-0.238, +0.429]$ (the opposite sign gives -0.238). Whether language priors carry ranking signal remains open.

Table 28: Frame confounds for the two frame-fed evaluators: the frames in order, shuffled, and the first frame alone.

evaluator	frames	τ	95% CI	winner
InternVL3-8B	8, in order	$+0.714$	$[+0.238, +0.905]$	`pg_diffusion`
InternVL3-8B	8, shuffled	$+0.714$	$[+0.238, +0.905]$	`pg_diffusion`
InternVL3-8B	first only	-0.524	$[-0.714, +0.429]$	`pi0`
SigLIP2-B/16	3, in order	-0.714	$[-0.905, -0.238]$	`pg_binning`
SigLIP2-B/16	3, shuffled	-0.714	$[-0.905, -0.238]$	`pg_binning`
SigLIP2-B/16	first only	$+0.524$	$[-0.048, +0.810]$	`pi0_fast`

Frame order and a single frame. The confound arms re-run the two *frame-fed* evaluators, which receive sampled frames rather than video, with the first frame only or with shuffled frames. For these two evaluators, shuffling frame order leaves the ranking unchanged although it changes 18.2% of InternVL3's scores (Table 28). SigLIP2's score is a mean over frames, so shuffling cannot change it, and its shuffled row checks the implementation. A single early frame removes agreement without reversing it: neither single-frame interval excludes zero. For SigLIP2 the reversal appears only with the later frames. One explanation consistent with this is that the cosine rewards frames that still look like the instruction's scene before the robot has disturbed it, and a policy that barely acts leaves the scene undisturbed. The control does not identify this mechanism.

I PUBLISHED CORRELATIONS

We collected every correlation between an evaluator or simulator and real outcomes that nine papers report as a result (Li et al., 2025; Quevedo et al., 2026; Jain et al., 2026; Tseng et al., 2026; Jeon et al., 2026; Li et al., 2026; Ma et al., 2026; GigaWorld Team et al., 2026; Wang et al., 2026a), 82 coefficients in total. Summary values over other coefficients, such as a worst case across environments, are not counted separately. For each we identified what it correlates across and assigned one of three classes (Table 30). Only a correlation across independent policies speaks to generalization across policies.

Two inferential questions. A correlation over policies asks whether the evaluator tracks real performance beyond the policies tested. Its effective sample size is the number of independent

policies. Our audit asks a different question. Given a fixed set of candidates, how uncertain is the selection among them because of finite evaluation data? Here the resolution comes from the comparisons. A small candidate set limits the first question but not the second, because a few fixed policies can be ordered confidently with enough comparisons. Coefficients pooled over task–policy cells still inform the order of the evaluated candidates, but they do not establish generalization to new policies.

What each paper claims. The papers use their coefficients for different claims (Table 29). Only RoboWorld states its result as a property of policy ranking in general: "world-model rollouts can approximate real-world policy rankings". Five papers test fixed candidates, tasks, or environments and describe the result in general terms, for example "a reliable proxy for real-world robot evaluation" (WorldEval). Three make no general claim. The sim-and-real recipe study, which supplies 22 of the 36 policy-level coefficients, uses them to compare simulators and post-training recipes, and states that its results "should not be interpreted as a definitive ranking of existing simulators or policies".

Our intervals measure how far a correlation over n independent policies speaks for policies beyond them. The uncertainty in ordering the tested candidates, or in a paired contrast between two simulators on the same policies, is a separate question. A few candidates evaluated with many trials can be ordered precisely, and showing it needs trial-level or task-level data that most papers do not print. The literature analysis therefore shows that many published coefficients give limited evidence for generalization to new policies, and that the reporting rarely quantifies uncertainty in the selection decision itself. It does not show that the tested candidates were ordered wrongly.

Table 29: What each paper uses its evaluator–reality correlations for, from the papers' own statements.

paper	what the correlation is used to show	general claim about ranking policies
WorldEval	the order of four policies; a proxy for real evaluation	in wording only
WorldGym	agreement across 17 tasks with three policies	in wording only
PolaRiS	policy ranking in unseen environments	in wording only
SC3-Eval	checkpoint ranking, including one held-out task	stated as a goal
RoboWorld	policy ranking in general	yes
dWorldEval	ranking across architectures and checkpoints	in wording only
PiL-World	relative performance of one policy's checkpoints	no
GigaWorld-1	agreement of a VLM judge with human raters	no
sim-and-real recipe	comparing simulators and post-training recipes	no, disclaimed

Policy-level (36). An ordinary correlation over $n \geq 4$ independent policies. We give a Fisher-z interval at n, built on the Fisher transform of the coefficient, which is roughly normal. For Spearman coefficients the interval uses the Fieller–Hartley–Pearson variance $1.06/(n-3)$ (Fieller et al., 1957). Where the source prints full rankings, we give a two-sided exact permutation p-value over all orderings instead. **Approximation (25).** A coefficient that is not a correlation over policies: an average of several per-task correlations (11), or a correlation pooled over task–policy cells that share policies (14). An interval computed at the number of policies is shown in Figure 1 as a sensitivity approximation but not counted, because the coefficient's target is not the policy ranking. WorldGym is an example. Its 51 printed pairs reproduce $r = 0.785$, and a bootstrap over its 17 tasks gives $[+0.681, +0.870]$, an interval about task–policy cells. As evidence about new policies, it rests on three policies, and judging the order of those three would need their per-trial outcomes. **No policy-level interval under this protocol (21).** The sample size is never stated (8), there are fewer than four policies (9), or the unit is not a policy (4).

At the level of papers, four of the nine (the sim-and-real recipe study, WorldEval, RoboWorld, and PolaRiS) report at least one policy-level coefficient whose interval excludes zero or whose permutation test gives $p \leq 0.05$. Several coefficients come from the same study, so the counts above are not independent failures. The count depends on the test direction: with one-sided tests at 0.05 (a 90% interval, or half the two-sided permutation p), 20 of the 36 cannot rule out zero.

Interval coverage at small n. We checked coverage by simulation (bivariate normal, 20,000 replicates per cell, $n = 4$ to 8, true correlation $\rho \in \{0, 0.5, 0.8, 0.9, 0.95\}$). The Pearson Fisher-$z$ interval covers 94.8–96.6% of the time in every cell, slightly conservative at $n = 4$. The Spearman

Table 30: Published coefficients by class, and what the policy-level ones support.

class	count	result
policy-level, Fisher-z interval	25	15 include zero; median width 1.06 of a possible 2
policy-level, exact permutation test	11	10 have $p > 0.05$
approximation (average or pooled)	25	shown, not counted
no policy-level interval here	21	—
total	82	25 of 36 policy-level coefficients are compatible with zero

interval is unreliable at $n \leq 5$, mostly because a sample often reaches a Spearman coefficient $|r_s| = 1$, which has no interval. For this reason, the policy-level Spearman coefficients at $n = 5$ carry exact permutation tests instead. The three Spearman intervals we do use are at $n = 8$, where coverage is 94.6–95.3% for $\rho \leq 0.9$. These coverage results apply to bivariate-normal data.

Recomputed from published data. All eleven Spearman coefficients of the sim-and-real recipe study (Wang et al., 2026a) reproduce exactly from its printed rankings. WorldEval's summary value 0.942 is the mean of three per-task correlations over the same four policies, with intervals $[-0.039, +0.999]$, $[-0.502, +0.998]$, and $[+0.325, +1.000]$. SC3-Eval's 0.929 is computed over 21 points that are three criteria on seven checkpoints. At the seven independent checkpoints, its interval is $[+0.586, +0.990]$ rather than $[+0.830, +0.971]$.

What $n = 7$ can resolve. We simulated evaluator scores with true correlation $r^* = 0.929$ against the reference strengths: for each n from 4 to 15 (10,000 iterations), draw n of the 15 RoboArena policies weighted by episode count, take one bootstrap draw of their strengths, and generate scores through a Gaussian copula (correlated normal draws) on van der Waerden normal scores (ranks mapped to normal quantiles). At $n = 7$ the central 95% of measured Pearson r spans $[0.51, 0.96]$ on raw strengths and $[0.86, 0.99]$ on normalised scores. The raw scale is wider because the last-placed policy is an outlier. Here 0.929 is the copula correlation of the simulated scores with the normal scores of the strengths. The simulation runs forward, from a true correlation to the spread of measured ones. It illustrates sampling variability at $n = 7$ and is not a confidence set for the true correlation behind an observed 0.929, which the intervals of Table 30 provide. The same pattern holds for the newest claim we found. RoboLab (Yang et al., 2026) reports Spearman $\rho = 1.00$ and Pearson $r = 0.68$ against RoboArena on four shared policies. A perfect ordering of four policies has exact one-sided permutation $p = 1/24 \approx 0.042$ (two-sided $1/12$). This is limited evidence whose strength depends on the test direction, and the Pearson interval is $[-0.81, +0.99]$.

J Pre-registration: hypotheses and outcomes

We registered six hypotheses, the policy set, the aggregation, the contamination and compliance rules, and the software environment before the full campaign. Small pilot runs on 64 episodes preceded the registration and informed its engineering choices (camera priority, frame caps, checkpoint conversion), not its hypotheses. Table 31 gives each outcome. One hypothesis resolved as predicted, one was falsified, one is inconclusive, one could not be measured on this data, and two were not run.

Changes after registration. Three changes were made after registration and before the affected outcomes were examined: the parse rule for tag-declaring evaluators (Appendix A), the activation of the confound arms, and the extension of the noise floor to every evaluator after InternVL3's kernel sensitivity appeared. Every decision-level analysis in the main text (winner agreement, regret, population targets, readout and aggregation controls, inclusion thresholds, the two added configurations) was added after registration and is exploratory (Table 1). The registered analysis named a Holm correction (a step-down adjustment for multiple tests) without p-values or an ordering, so it could not be executed as written. Testing each registered evaluator's τ against zero with bootstrap p-values and a Holm adjustment rejects zero for all six at $\alpha = 0.05$ (adjusted p from 0.0024 to 0.0032). H5 was registered over the judges on RoboRewardBench and run over the three that we audit.

Table 31: Registered hypotheses and outcomes.

	prediction	outcome
H1	evaluators name different winners	**confirmed**: the six evaluators name three different winners, and four of six name a winner other than the reference's.
H2	a no-pixel evaluator retains $\geq$ half the video τ	**not measurable**: both sides of a session share one instruction, so blind prompts are identical.
H3	the n=7 interval for Pearson r contains 0.5	**falsified**, narrowly: the raw-scale band starts at 0.51.
H4	some evaluator scores injected bad episodes above the median	**not run**.
H5	benchmark MAE order disagrees with ranking order	**inconclusive**: MAE varies 1.8$\times$ across three judges whose τ cannot be separated.
H6	downstream correlations of world-model judges span ≥ 0.4	**not run**: it needs regenerated videos.

Claims excluded by the registration. The registration excludes claims about evaluators or policies we did not test, claims about running an evaluator in a closed control loop, causal claims beyond what the confound arms show, and rankings of the audited evaluators against one another on τ. The paper follows these limits.

K Extended related work

Automatic evaluators for robot policies. World models generate an episode, and human raters or a model judge success on the generated rollout (Li et al., 2025; Quevedo et al., 2026; Guo et al., 2026; Tseng et al., 2026; Jeon et al., 2026; Li et al., 2026; Ma et al., 2026; GigaWorld Team et al., 2026). Reward models score a real episode directly: RoboReward (Lee et al., 2026) releases 4B and 8B checkpoints and the RoboRewardBench leaderboard, scored by mean absolute error against human-verified progress labels. Concurrent work (Wang et al., 2026c) critiques its 1–5 scores. LLM-as-a-Verifier (Kwok et al., 2026) reads a continuous score from token probabilities and reports 87.4% pairwise accuracy on RoboRewardBench with Qwen3.6-35B. General LLMs, and VLMs in the same way, are prompted as judges (Zheng et al., 2023), and vision-language similarity serves as a zero-shot reward (Rocamonde et al., 2024; Sontakke et al., 2023). R2S-Eval (Wang et al., 2026b), concurrent with this work, has eight open-weight VLMs compare rollout pairs in calibrated simulation and aggregates the preferences with Bradley–Terry over six policies, the aggregation we audit. Simulation suites validate the same way: RoboLab (Yang et al., 2026) checks itself against RoboArena on four shared policies. Most of these systems validate by correlating their scores with real outcomes over three to eight policies or checkpoints.

Human ground truth for robot policies. RoboArena (Atreya et al., 2025) runs distributed double-blind A/B evaluation on DROID hardware and releases the raw comparisons. AutoEval (Zhou et al., 2025) automates real-robot evaluation with success detection and scene resets on WidowX arms. Tian et al. (2026) test three reward models on RoboArena labels and find preference-ordering accuracy falling from 0.72–0.77 on pick-and-place to 0.52–0.62 on tool use. They study trajectory pairs, not policy selection. ArmnetBench (Selvaraj et al., 2026) adds 2,518 human-scored policy rollouts as a second dataset.

Auditing evaluators elsewhere. LLM judges disagree with humans, prefer their own generations (Panickssery et al., 2024), are sensitive to candidate order (Zheng et al., 2023; Wang et al., 2024), and face a proven ceiling as substitutes for human labels (Dorner et al., 2025). Tool-use audits find substantial evaluator–human disagreement (Bhat et al., 2026). Methodology papers urge authors to report variance, standard errors, and sample size (Dodge et al., 2019; Bouthillier et al., 2021; Miller, 2024). Numerical nondeterminism in LLM inference is documented for text generation (Yuan et al., 2025; He & Thinking Machines Lab, 2025). Embodied evaluators add video decoding, frame sampling, and camera choice, which can change the visual evidence an evaluator sees. Our implementation controls hold decoded pixels and all other inputs fixed.

L RELEASED ARTIFACTS

The anonymous repository `https://anonymous.4open.science/r/evaluator-selection-audit/` contains: the audit suite (evaluator runner, video decoding, induced-preference Bradley–Terry aggregation, paired and cluster bootstraps, and every analysis script, with its tests); the registered protocol; the frozen reference ranking with its 10,000 bootstrap draws, pairwise ordering probabilities, and rank-exchange matrix; per-episode scores and raw model output for every evaluator and arm; and the coefficient-level literature table with the unit of analysis and interval method for each of the 82 coefficients. Raters and sites carry stable pseudonyms. A `README` names the script behind each main result, and each script documents its own command. The report-card script (Section 5) runs any evaluator's per-episode scores through the full audit. Cards for all evaluators in this paper are included.

Reading a card. The card answers three questions separately. *What would I select now?* The frozen full-data pick. *What could that choice cost under the named reference?* Its frozen-choice regret, read as a preference probability through $1/(1+e^{-\Delta})$, and its held-out regret against a random pick. *How stable is the procedure under resampling?* Winner agreement. RoboReward-8B on the matched cohort shows why the three must stay apart. It picks the reference winner (frozen-choice regret 0.002, so the reference prefers its winner with probability 0.50). Among the six competitive policies, its held-out pick gives up 0.136, against 0.252 for a random pick. Its winner agreement is only 0.419 $[0.07, 0.91]$. The decision is good, the margin over chance among close candidates is modest, and the procedure is unstable because the top two policies are close. If sites or raters are counted equally, the reference target changes while the pick stays the same, and the pick gives up 0.14 to 0.28, a preference probability of at most 0.57.