When the Safety Layer Is Not Safe: In-depth Analysis of VLA Failure Detection under Distribution Shift

Anonymous authors
Paper under double-blind review

Abstract

Vision-language-action (VLA) policies achieve impressive success rates on manipulation benchmarks, yet recent robustness analyses show this competence is brittle and collapses under modest open-world variation. Runtime failure detectors are increasingly used alongside VLA policies. Like the policies they monitor, these detectors perform well under ideal conditions, raising an alarm in time for the robot to stop, back off, or ask for help. Existing work, however, trains, calibrates, and evaluates these detectors only on clean rollouts and rarely tests them under the distribution shifts (e.g., a changed camera viewpoint or robot initial state) that cause policies to fail in the first place. We therefore collect a 5,501-rollout corpus of successes and failures of two VLA policies under seven controlled perturbation dimensions and introduce a three-distribution protocol that separates a detector's ranking ability from the validity of its calibrated threshold. We find that thresholds break before rankings: clean-calibrated detectors retain most of their discrimination under shift but read unfamiliar success as failure, raising false alarms by 16.5 percentage points on the shifts to which policies are most sensitive. The direction of the break is set by how the score accumulates signed per-step evidence, not by how far features drift: on the same features, kNN detectors over-alarm while a Mahalanobis detector falls silent and misses failures. Task novelty, by contrast, mainly erodes ranking. No evaluated repair restores nominal calibration without sacrificing detection, and label-free recalibration is circular because the unlabeled deployment pool contains the failures the monitor is meant to catch. Neither a high AUROC nor a low false-alarm rate therefore certifies a monitor. A safety layer built for the moment a policy cannot be trusted must be evaluated and calibrated under the shifts that create that moment, with scores that do not mistake novelty for failure. Code and data will be released upon publication.

1 Introduction

Vision-language-action (VLA) policies now reach high success rates on multitask manipulation benchmarks (Brohan et al., 2023; Kim et al., 2024; 2025; Black et al., 2024). This competence is brittle. Controlled robustness studies show that modest changes in camera viewpoint, robot initial state, lighting, or instruction wording can collapse success rates that appeared saturated (Fei et al., 2025; Zhou et al., 2025). Runtime failure detectors have therefore become standard companions to learned robot policies, VLA policies included (Agia et al., 2024; Xu et al., 2025; Römer et al., 2026a; Gu et al., 2026). They read signals from the policy while it acts and raise an alarm early enough for the robot to stop, back off, or ask a human for help. Such a detector matters most when the policy it monitors is about to fail, and for VLA policies that moment usually arrives with a change in the world.

The way these detectors are built does not reflect this. SAFE (Gu et al., 2026), for example, trains lightweight probes on the hidden states of the policy and calibrates a time-varying alarm band on successful rollouts with conformal prediction, and FAIL-Detect and FIPER likewise set their thresholds from nominal successful executions (Xu et al., 2025; Römer et al., 2026a). Evaluation then reports AUROC and alarm rates on held-out rollouts from the training conditions or on held-out tasks. Training, calibration, and evaluation thus all take place on clean rollouts, and the detector

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.

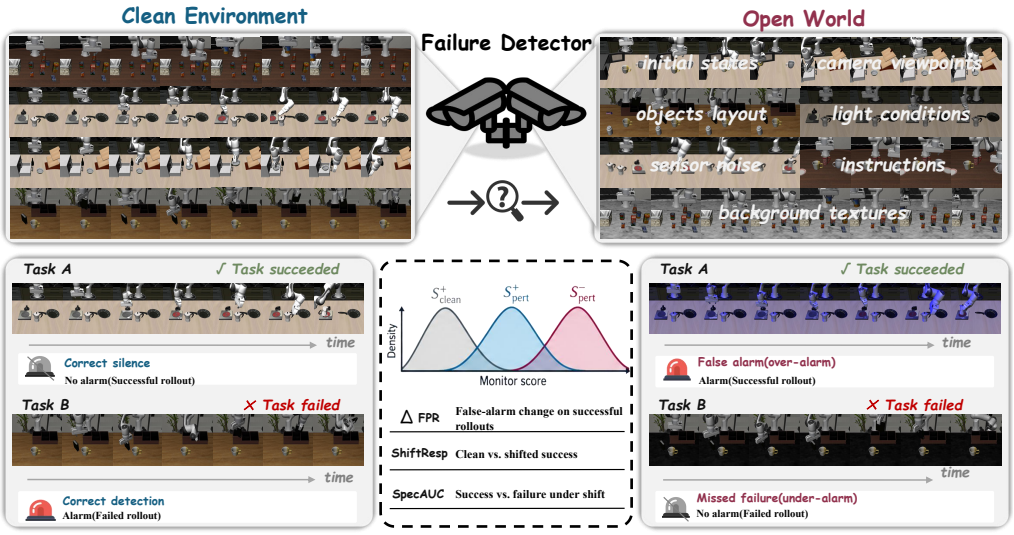

Figure 1: **A failure detector must stay reliable when the environment of its policy changes.** The top row contrasts the clean environment in which a detector is calibrated with the shifted open world in which it is deployed. The bottom row shows correct silence and correct detection under clean conditions, and a false alarm and a missed failure under shift. The center panel shows the three score distributions that our protocol compares and the metric that separates each pair. The decisions are illustrative rather than measured.

rarely meets, in a controlled way, the shifts that make its policy fail. Under such a shift a detector can go wrong in two distinct ways. Its scores may still rank failures above successes while its clean threshold no longer means what it meant, so that successful but unfamiliar executions trigger alarms. Or its scores may drift away from the alarm region, so that failures pass in silence. A single AUROC or a single false-alarm rate cannot tell these cases apart, and they call for different remedies.

Concurrent work has begun to make detectors more robust. SAFECAST (Rajaprakash et al., 2026) adds contrast sets of perturbed rollouts to probe training and calibration, and perturbation-based uncertainty (Lee & Har, 2026) derives failure signals from perturbations of the model itself. What is still missing is a diagnosis of how clean-calibrated detectors fail under shift. Which part of the detector breaks, in which direction, and can it be repaired with the information a deployment has?

This paper provides that diagnosis. We collect a 5,501-rollout corpus that contains both successes and failures of two primary policies, OpenVLA-OFT (Kim et al., 2025) and π_0 (Black et al., 2024), under controlled LIBERO-plus perturbations along seven dimensions (Fei et al., 2025). Collection targets at least 30 successes and 30 failures for each policy and dimension, because a detector cannot be evaluated where one outcome is missing. On this corpus we introduce a three-distribution protocol (Figure 1) that compares clean successes, perturbed successes, and perturbed failures. The comparison separates three questions that a clean evaluation conflates. Does the clean threshold still hold on successful executions? Does the score move with novelty alone? Does it still rank failures above successes under the new conditions?

Four findings emerge. *Thresholds break before rankings.* On camera and robot-initialization shifts, the two factors to which VLA policies are most sensitive (Fei et al., 2025), the SAFE probes keep most of their discrimination yet raise false alarms on successful rollouts by 16.5 percentage points on average and by up to 31.3. *The direction of the break is not fixed.* On the same features, kNN detectors over-alarm by 44 to 50 points while a Mahalanobis detector falls silent and ranks shifted failures no higher than shifted successes. A temporal audit traces this direction to the signed evidence that each score accumulates over time, which feature drift alone does not predict. *Task novelty and condition shift damage different facets.* Across all 120 seen and unseen task splits, novel tasks cost far more ranking than condition shift, while condition shift displaces the operating point even where ranking survives. *Repair is bounded by information.* Label-free recalibration is poisoned by the

failures in the deployment pool, with drift-estimation bias of 48% to 317%, and every evaluated repair that reduces calibration error gives up failure detection. Our contributions are threefold.

- A two-outcome corpus under controlled shifts and a three-distribution protocol that make the reliability of a failure detector under distribution shift measurable by separating threshold validity, novelty response, and failure discrimination.
- An empirical account of where, in which direction, and why clean-calibrated detectors break, including a temporal mechanism for the direction of miscalibration and an exhaustive factorial that separates task novelty from condition shift.
- A repair study indexed by available information, which exposes calibration circularity and yields deployment decision rules. Code and the rollout corpus will be released.

2 Related work

Runtime failure detection for robot policies. Runtime monitors for learned policies detect failure from temporal inconsistency and lack of progress (Agia et al., 2024), from the uncertainty of generative policies (Xu et al., 2025; Römer et al., 2026b), from observation novelty combined with action uncertainty (Römer et al., 2026a), from conformal monitoring of actions and states (Chen et al., 2026), from perturbations of the model (Lee & Har, 2026), or from learned probes on VLA hidden states (Gu et al., 2026). SAFECAST (Rajaprakash et al., 2026) trains and calibrates such probes with perturbed contrast sets, including analyses by perturbation family. We do not propose a new detector. We ask how the ranking and the calibrated threshold of existing detectors behave under controlled shift, resolve the answer by perturbation dimension, cross it with task novelty, and test which repairs the available information supports.

Failure detection versus out-of-distribution detection. In classification, failure detection and out-of-distribution detection respond to different properties of an input and can disagree (Hendrycks & Gimpel, 2016; Jaeger et al., 2022), and uncertainty estimates that are calibrated in distribution degrade under dataset shift (Ovadia et al., 2019). Our $\mathrm{ShiftResp}$ metric carries this distinction to sequential robot monitoring, where a detector that responds to benign novelty raises alarms on successful but unfamiliar executions.

Robustness benchmarks and calibration under shift. LIBERO (Liu et al., 2023) benchmarks multitask manipulation, and LIBERO-plus and LIBERO-PRO add controlled perturbations and generalization tests (Fei et al., 2025; Zhou et al., 2025). These benchmarks evaluate the policy. We use their perturbations to evaluate the monitor as well, which requires both outcomes within every perturbation dimension. Conformal prediction gives marginal guarantees under exchangeability (Vovk et al., 2005; Angelopoulos & Bates, 2023), with functional extensions for trajectories (Diquigiovanni et al., 2022) and variants for covariate shift, bounded shift, non-exchangeable data, and online feedback (Tibshirani et al., 2019; Cauchois et al., 2024; Barber et al., 2023; Gibbs & Candes, 2021). These methods motivate our repair families. Their guarantees rest on assumptions such as known likelihood ratios, a bounded divergence, or prompt feedback, which a shifted deployment stream with unknown outcomes rarely meets, and our repair study measures what happens then.

3 A three-distribution protocol

3.1 Policies, controlled shifts, and corpus

We study two VLA policies with different action heads, OpenVLA-OFT (Kim et al., 2025) with parallel decoding and an L1 regression head, and π_0 (Black et al., 2024) with a flow-matching head, both from their LIBERO checkpoints. Shifts come from LIBERO-plus (Fei et al., 2025), applied to the ten LIBERO-10 tasks along seven dimensions (camera viewpoint, robot initial state, language instruction, object layout, lighting, background texture, and sensor noise), each at five severities from L1 to L5. Layout-L5 and light-L1 have no source variants, and layout-L4 has limited coverage and is treated as exploratory.

A detector evaluation needs both outcomes in every condition, which the success rate of a policy alone does not supply. We therefore allocate rollouts adaptively from a pilot, targeting at least 30

successes and 30 failures for each policy and dimension. The perturbed primary corpus contains 1,091 OFT and 890 π_0 rollouts, 594 of which are failures. All fourteen policy and dimension cells meet the success target, and all but OFT background and OFT language, with 18 and 11 failures, meet the failure target. Each policy also has 500 clean rollouts from the same software stack, which gives 2,981 primary trajectories. The full 5,501-rollout inventory adds two auxiliary policies and legacy clean rollouts that serve only as regression checks (Appendix A). Because allocation is adaptive, failure fractions describe the corpus rather than deployment prevalence.

3.2 Detectors and calibration

Our main detectors are the SAFE MLP and LSTM probes (Gu et al., 2026). Clean cross-validation within the training data selects the `Last` feature aggregation for OFT and `First&Last` for π_0. To test whether conclusions depend on how a score is built, we add four detectors on the same features, namely Mahalanobis (Lee et al., 2018), Euclidean and cosine kNN (Sun et al., 2022), and LogpZO (Xu et al., 2025). Our distance detectors are supervised contrasts. Each step contributes its distance to the success reference minus its distance to the failure reference, and the score sums these signed increments over time. The MLP likewise sums nonnegative per-step outputs, so its score never decreases, whereas the LSTM emits a bounded score at each step (Appendix B).

For each of five seeds, the clean corpus is split $250/125/125$ into training, calibration, and evaluation sets, stratified by task and outcome. As in SAFE, only calibration successes set the time-varying band δ_t, and a rollout raises an alarm at the first step with $s_t > \delta_t$. Our implementation takes a finite-sample-adjusted quantile of the calibration scores separately at each timestep (Equation 5). Such a pointwise band does not bound the probability that a successful rollout crosses it somewhere, so its clean false-alarm rate can differ from α. We therefore treat α as a nominal setting and measure every shift effect relative to the false-alarm rate on held-out clean successes, which removes the clean miscalibration of the band itself from the comparison.

3.3 Separating ranking from threshold validity

For a perturbation dimension d, let $\mathcal{S}^+_{\text{clean}}$, $\mathcal{S}^+_{\text{pert}}(d)$, and $\mathcal{S}^-_{\text{pert}}(d)$ denote held-out clean successes, perturbed successes, and perturbed failures, and let $m = \max_{t \le T} s_t$ be the peak score of a rollout. We compute

$$\Delta\text{FPR}_d(\alpha) = \text{FPR}_\alpha[\mathcal{S}^+_{\text{pert}}(d)] - \text{FPR}_\alpha[\mathcal{S}^+_{\text{clean}}], \tag{1}$$

$$\text{SpecAUC}_d = \Pr(m_d^- > m_d^+) + \tfrac{1}{2}\Pr(m_d^- = m_d^+), \tag{2}$$

$$\text{ShiftResp}_d = \Pr(m_d^+ > m_{\text{clean}}^+) + \tfrac{1}{2}\Pr(m_d^+ = m_{\text{clean}}^+). \tag{3}$$

The false-alarm change ΔFPR asks whether the clean threshold still holds on successful executions. The within-shift AUROC SpecAUC compares failures with successes under the shifted condition and asks whether the ranking survives. The shift response ShiftResp compares successes with successes and asks whether the score moves with the change in conditions alone. A shift-robust failure detector keeps SpecAUC high, ShiftResp near 0.5, and ΔFPR near zero. A ShiftResp above 0.5 means that the detector partly acts as a novelty detector and reads unfamiliar success as failure, and a ShiftResp below 0.5 means that its scores move away from the alarm region. All alarms are computed offline on recorded executions, mechanism claims are restricted to the audited detectors, and bootstrap units and the treatment of repeated seeds are specified in Appendix B.

4 Thresholds break before rankings

On camera and robot-initialization shifts, the SAFE probes raise false alarms on successful rollouts by 16.5 percentage points on average across two probes, two policies, and $\alpha \in \{0.1, 0.2\}$ (Figure 2a). The increase reaches 31.3 points for the OFT MLP on camera at $\alpha = 0.2$. These are the two factors to which VLA policies are most sensitive (Fei et al., 2025), so the monitor degrades exactly where it is needed most. The mean ShiftResp of these probes is 0.61. Successful executions under shift therefore outscore clean successes, and the clean threshold reads unfamiliar success as failure.

Most of the ranking survives nonetheless (Figure 2b). The OFT MLP reaches a SpecAUC of 0.85 on camera and 0.87 on robot initialization against a clean AUROC of about 0.90, and the π_0 MLP

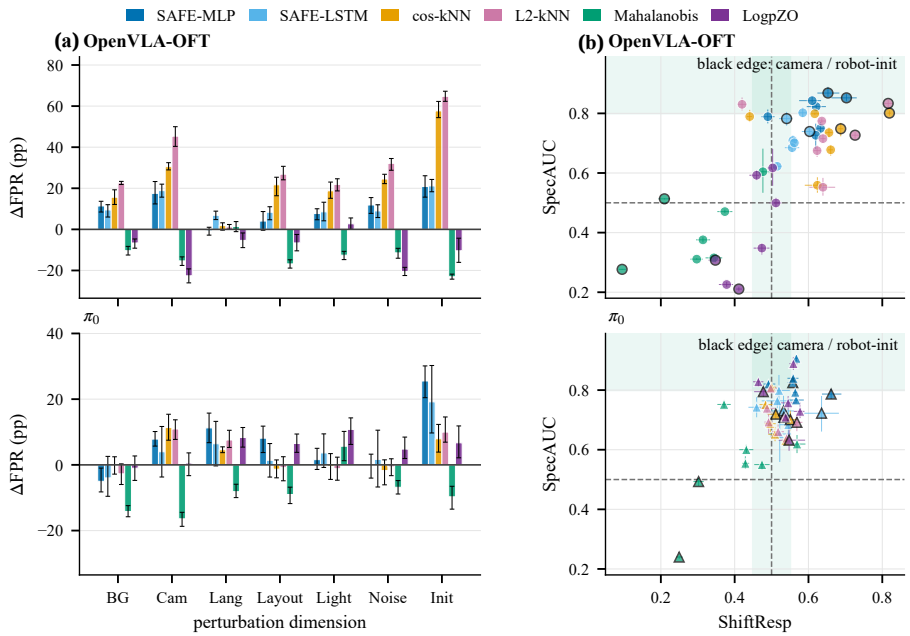

Figure 2: **False-alarm change and failure discrimination capture different effects of shift.** Panel a shows $\Delta\mathrm{FPR}$ at $\alpha = 0.1$ by perturbation dimension and detector, with OFT and π_0 in separate rows on different vertical scales. Panel b plots $\mathrm{SpecAUC}$ against $\mathrm{ShiftResp}$ on shared scales, with circles for OFT and triangles for π_0. Black outlines mark camera and robot initialization. Shading marks $\mathrm{ShiftResp}$ near 0.5 and $\mathrm{SpecAUC} \geq 0.8$, the region of a shift-robust failure detector. Error bars are bootstrap intervals over five seed-level estimates.

reaches 0.83 and 0.79. The monitor still knows which rollouts look worse. What it has lost is where to draw the line. The gap is sharpest at individual severities. At robot-init L5 the OFT MLP keeps an AUROC of 0.93 while only 20% of successful rollouts escape an alarm at a nominal coverage of 90%, and at robot-init L2 coverage is already 0.52 for OFT and 0.44 for π_0 (Appendix C). The size of this move tracks hidden-state drift. Across seed and severity cells, median Mahalanobis feature drift correlates with MLP false-alarm inflation at Spearman $\rho = 0.64$ for OFT and 0.38 for π_0. As Section 5 shows, drift does not predict the direction of the move.

The effect also depends on the policy. Language perturbations leave the OFT MLP essentially unchanged ($\Delta\mathrm{FPR} = -0.7$ points and $\mathrm{ShiftResp} = 0.49$) but move the π_0 MLP ($+11.3$ points and 0.57), and π_0 barely registers background changes. A perturbation displaces a probe only insofar as it displaces the hidden states of the policy. The OFT result is consistent with the weak instruction sensitivity reported for VLA policies (Fei et al., 2025), although aggregate scores alone cannot establish that a policy ignores instructions. Per-dimension values are in Appendix C.

What this looks like on a robot. Figure 9 in Appendix E instantiates the protocol on recorded executions of one instruction. For OFT the peak scores keep the ordering clean success $<$ shifted success $<$ shifted failure ($2.10 < 5.34 < 9.76$), yet the shifted success crosses the clean band at feature step $t = 4$ and still completes the task, as does the π_0 shifted success after alarming at $t = 15$. The cases are selected for illustration, and the appendix documents their selection and time alignment.

5 Which way thresholds break depends on score accumulation

Over-alarming is not the only way in which a clean-calibrated monitor fails. Averaged over the two strong shifts and $\alpha \in \{0.1, 0.2\}$, Euclidean and cosine kNN raise OFT false alarms by about 50 and 44 points, far more than the SAFE probes (Figure 2a). Mahalanobis moves the other way and lowers

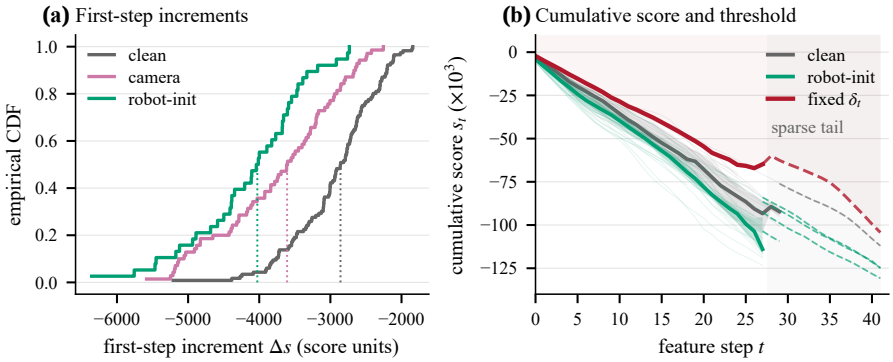

Figure 3: **Signed per-step evidence moves the Mahalanobis monitor away from its alarm band.** Panel a shows OFT seed-0 empirical CDFs of first-step Mahalanobis increments for 116 clean, 70 camera, and 38 robot-initialization successes, with dotted ticks at the medians. Panel b shows 30 clean and 30 robot-initialization score paths with median and interquartile summaries where at least six paths remain. The red curve is the fixed band, with the alarm region above it. Dashed tails indicate sparse support, including fewer than 20 calibration successes.

false alarms by 26 points for OFT and by 16 for π_0, and LogpZO also moves downward for OFT. Fewer false alarms would be welcome if failures were still caught, but they are not. On camera and robot initialization, the SpecAUC of Mahalanobis falls to about 0.5 or below for both policies, as does that of LogpZO for OFT (Figure 2b). Shifted failures then score no higher than shifted successes, which themselves now sit further below the band, so a band that stops firing on successes largely stops firing on failures as well. Figure 4 shows the consequence on two recorded failed executions scored by three detectors on the same features. On an OFT book-to-caddy failure under robot-init L3, the SAFE-MLP first alarms at feature step $t = 5$ and L2-kNN at $t = 1$, whereas Mahalanobis stays below its band to the end of the scoring window; on a π_0 mug-to-microwave failure under robot-init L5 the crossings come at $t = 13$ and $t = 11$, again with no Mahalanobis crossing. For a safety layer this silence is the more dangerous failure, because nothing signals that anything has gone wrong. Both cases are selected for illustration (Appendix E).

Feature displacement alone does not explain why detectors that read the same features move in opposite directions. When we inject synthetic mean shifts of 0.5, 1.0, and 2.0 trajectory-time means into clean evaluation features, all three distance detectors raise their false alarms, Mahalanobis included (Appendix D). The direction under real shifts must therefore come from how each score converts features into per-step evidence and accumulates it. The Mahalanobis score sums the distance to the success reference minus the distance to the failure reference. Under real shifts both distances grow, but the distance to the failure reference grows faster, so the increments become more negative. Figure 3a shows this from the first step, where the median OFT increment moves by about -800 score units under camera shift and by about $-1{,}200$ under robot initialization. Accumulation then carries each trajectory further below the fixed band at every later step (Figure 3b), and the median gap below the band widens in all 20 policy, dimension, and seed comparisons. Successful trajectories are about equally long with and without shift (27.34 versus 27.17 feature steps for OFT), which argues against a simple duration effect.

Accumulation has no direction by itself, since the kNN scores also sum signed contrasts and drift upward. What accumulation does is turn a small change in the sign or size of each increment into a displacement that grows with time against a band calibrated step by step on clean data. The same temporal lens explains part of the over-alarming of the MLP. Replacing the first quarter of the score increments of a shifted OFT rollout with clean increments from the same task removes 30.7%, 34.7%, and 18.7% of alarms under camera, light, and robot-init shifts and delays the first crossing by a median of 7.5 to 10 feature steps, whereas the same splice barely moves π_0 (Appendix D). For OFT, an early offset that the monotone MLP score never forgets accounts for a large share of the false alarms. In short, feature drift tracks how far an operating point moves, while the sign and accumulation of per-step evidence decide which way. A detector can look calm under shift simply

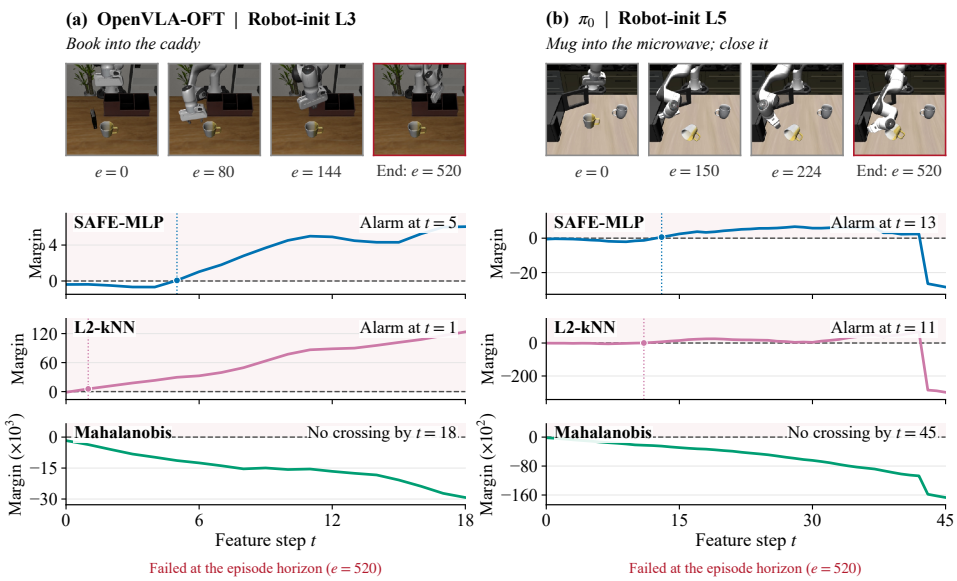

Figure 4: **A failed robot execution can raise an alarm under one detector and stay below the band of another.** Below the agent-camera frames of two unsuccessful executions, three detectors score the identical recorded prefix. Curves show $s_t - \delta_t$, the dashed zero line is the alarm boundary, and dots mark first crossings. Vertical axes keep detector-specific units (note the powers of ten), so magnitudes are not comparable across detectors; the final frame lies beyond the scoring window.

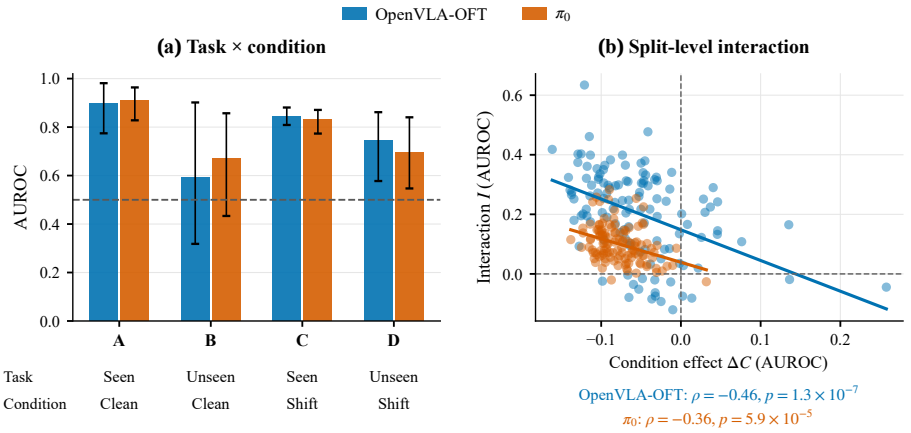

Figure 5: **Task novelty and condition shift affect the MLP monitor differently.** Panel a shows AUROC over all 120 splits. Bars are split means, and error bars span the central 95% of split-level values rather than a confidence interval for the mean. Panel b plots $I = D - B - C + A$ against $C - A$, where a positive I indicates a smaller combined loss than the additive prediction. The term $C - A$ is shared by both axes, which limits a mechanistic reading of their association.

because its score has drifted away from the band, so a falling false-alarm rate must always be read together with discrimination under the same shift.

6 Task novelty and condition shift damage different facets

SAFE was designed for multitask transfer and evaluated on held-out tasks (Gu et al., 2026). Deployment combines task novelty with condition shift, so we cross the two. For each of the $\binom{10}{3} = 120$ divisions of LIBERO-10 into seven seen and three unseen tasks, we train the MLP and fit its band

on seen-task clean data over five seeds, 1,200 trainings in total. We evaluate seen tasks under clean conditions (A), unseen tasks under clean conditions (B), seen tasks under perturbation (C), and unseen tasks under perturbation (D). For any metric, the interaction $I = D - B - C + A$ measures the departure from additive effects.

Task novelty is the main threat to ranking. The AUROC change $B - A$ is -0.31 for OFT and -0.24 for π_0, against $C - A = -0.05$ and -0.08 for condition shift (Figure 5a). Both axes threaten the threshold. For OFT, unseen-task clean successes alarm at 42.5% under nominal $\alpha = 0.1$ against 19.5% for seen-task clean successes, and condition shift adds the displacement documented in Section 4 even where ranking holds. The two losses are sub-additive in AUROC, with mean interactions of $+0.20$ for OFT and $+0.10$ for π_0. The negative split-level association in Figure 5b is consistent with saturation, but its axes share $C - A$ and so do not provide independent evidence.

Cell D is the closest proxy for deployment within our grid, and it is sobering. At $\alpha = 0.1$, alarms fire on 50.8% of successful OFT rollouts and on 40.8% of successful π_0 rollouts, while AUROC is 0.74 and 0.69. The choice of held-out tasks also matters. The B-cell AUROC of OFT ranges from 0.20 to 0.92 across the 120 splits, and the mean over three randomly chosen splits has a 95% range of 0.43 to 0.76. Conclusions drawn from a few task splits can therefore reflect which tasks were held out rather than the detector itself. Appendix F adds the coverage interaction, an LSTM comparison, and a failure-count dose response.

7 Repair is bounded by the information available

Because rankings survive where thresholds break, the natural remedy is to recalibrate rather than retrain. What recalibration can use depends on what a deployment observes, so we index repairs by information budget (Table 1). With clean calibration data only, we inflate the calibration quantile (Robust CP), following conservative conformal calibration (Cauchois et al., 2024). With immediate outcome feedback, we adapt the miscoverage level online (ACI), following adaptive conformal inference (Gibbs & Candes, 2021). With unlabeled deployment rollouts, we either re-anchor the band to the observed score drift or reweight calibration scores by estimated density ratios (Weighted FCP), following conformal prediction under covariate shift (Tibshirani et al., 2019). With unlabeled internal features, we align feature statistics with CORAL (Sun & Saenko, 2016). With labeled shifted rollouts, we add them to probe training or calibration (TrainAug and CalAug) in the style of SAFECAST (Rajaprakash et al., 2026). These are the concrete rules in our code. They adapt the cited ideas rather than reproduce the cited methods with their guarantees (Appendix G). Methods that use deployment data draw on a shared auxiliary pool of 155 language-L1 and layout-L1 rollouts, disjoint from the 1,826 perturbed evaluation rollouts.

Calibration circularity. Label-free recalibration runs into a circularity. A band must be calibrated on successes, yet an unlabeled deployment pool also contains the failures that the monitor exists to catch, and telling the two apart is the job of the monitor itself. Re-anchoring shows the consequence (Figure 6a). Relative to a diagnostic oracle that uses only the successes in the pool, its median drift-estimation bias ranges from 48% to 317% across four policy and pool cells whose failure fractions range from 6% to 31%. The bias is not monotone in contamination, so the failure fraction alone does not determine it. Keeping only the lowest-scoring 50% to 70% of trajectories and trimming the estimates still fails our bias criterion in all four cells, and estimated density-ratio weighting fails its criterion on all seven dimensions for both policies (Appendix G).

A trade-off rather than a fix. No evaluated repair combines low coverage error with high failure detection (Figure 6b), where coverage error is the mean absolute deviation from nominal coverage across evaluated dimensions. Quantile inflation comes closest to nominal, with 7 to 12 points of error, but its failure TPR falls to about 36%. All 100 audited selected bands equal the per-timestep maximum of the clean calibration scores, so this family has saturated, and static bands could be raised further only by giving up still more detection. Feedback adaptation reaches 11 to 15 points of error, CORAL 17 to 31 points with reduced detection, and labeled augmentation improves the trade-off for some π_0 settings without improving OFT uniformly. Along the envelope of tested operating points, every reduction in coverage error costs failure detection. Table 1 turns these results into decision rules indexed by the information that a deployment has.

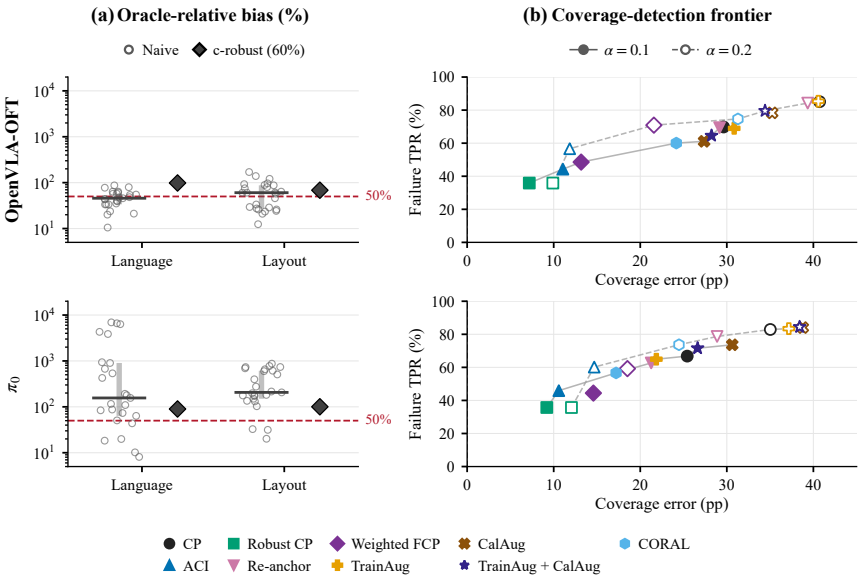

Figure 6: **Contaminated calibration pools and the empirical repair trade-off.** Rows correspond to OFT and π_0. In panel a, open circles are naive re-anchoring configurations, bars summarize medians and interquartile ranges, and diamonds show robustified re-anchoring at 60% retention. The red line marks 50% oracle-relative bias. In panel b, color and shape identify nine methods, filled and hollow markers denote $\alpha = 0.1$ and 0.2, and lines connect empirical envelopes. Values aggregate all evaluated perturbation dimensions. CP is the unrepaired clean band, and Robust CP, ACI, and Weighted FCP denote the adaptations specified in Appendix G.

Table 1: **Deployment decision rules indexed by available information.** Errors are mean absolute coverage deviations across evaluated perturbation dimensions. Entries summarize tested settings rather than certified guarantees.

Available information	Evaluated repair	Observed result and decision rule
Clean calibration only	Quantile inflation	7 to 12 points of error at about 36% TPR, with bands saturated at the clean maxima. Measure false alarms under shift instead of trusting α.
Immediate outcome feedback	ACI-inspired update	11 to 15 points of error, sensitive to arrival order and step size. Use only when every outcome is labeled promptly.
Unlabeled rollout scores	Re-anchoring and density-ratio weighting	Pool failures bias re-anchoring by 48% to 317%, and weighting fails every criterion. Avoid without a trusted success filter.
Unlabeled internal features	CORAL alignment	17 to 31 points of error with reduced TPR. Feature alignment alone does not restore calibration.
Labeled shifted rollouts	TrainAug and CalAug	Gains for some π_0 settings but not uniformly for OFT. The most informative budget, to be validated per policy.

8 Conclusion

A runtime failure detector matters most when a deployed policy cannot be trusted, and for VLA policies that moment usually arrives with a change in the world. Under such changes, clean-calibrated detectors keep much of their ranking but lose the meaning of their thresholds, drift toward over-alarming or silence depending on how their scores accumulate evidence, lose ranking mainly to task novelty, and resist label-free repair. A safety layer for open-world robots has to be evaluated under the conditions that motivate it, and the three-distribution protocol makes that evaluation routine. Appendix I discusses implications for deployment and the limitations of this study.

Ethics statement

This work studies safety monitors for robot policies entirely in simulation and involves no human subjects or personal data. Its findings argue against treating clean-calibrated monitors as certified safety mechanisms under distribution shift, and measured operating points from simulation should not be read as guarantees for physical deployment.

Reproducibility statement

Appendix A documents the corpus: policies, perturbation grid, blocked cells, the adaptive allocation rule, and the live checks. Appendix B specifies score semantics, the implemented calibration band, the operating grid, and the resampling unit behind every reported interval. Appendices D, F, and G specify the prefix-replay rule, the task-split design, and each repair as implemented together with its information budget. Appendix H records pinned software versions, revision-pinned checkpoints, seeds, and compute. Every reported number is indexed in a 545-entry provenance ledger that links it to its source artifact and verification record. Collection and analysis code, the rollout corpus, the machine-readable result tables, including the complete detector table over six values of α and all repair sweeps, and the provenance ledger will be released upon publication.

AI use statement

Large language models and coding agents assisted experiment execution, verification, and writing. All experimental rulings and every reported number were reviewed by the authors against the numerical provenance ledger. AI tools also assisted literature verification, code inspection, manuscript revision, and layout checks. The authors take full responsibility for the final text, experimental claims, and artifacts.

References

Christopher Agia, Rohan Sinha, Jingyun Yang, Zi-ang Cao, Rika Antonova, Marco Pavone, and Jeannette Bohg. Unpacking failure modes of generative policies: Runtime monitoring of consistency and progress. *arXiv preprint arXiv:2410.04640*, 2024.

Anastasios N Angelopoulos and Stephen Bates. Conformal prediction: A gentle introduction. *Foundations and Trends in Machine Learning*, 16(4):494–591, 2023.

Rina Foygel Barber, Emmanuel J Candes, Aaditya Ramdas, and Ryan J Tibshirani. Conformal prediction beyond exchangeability. *The Annals of Statistics*, 51(2):816–845, 2023.

Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, et al. backslash pi0: A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 1, 2024.

Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Choromanski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. *arXiv preprint arXiv:2307.15818*, 2023.

Maxime Cauchois, Suyash Gupta, Alnur Ali, and John C. Duchi. Robust validation: Confident predictions even when distributions shift. *Journal of the American Statistical Association*, 2024.

Lingling Chen, Zongyao Lyu, and William J Beksi. Reconvla: An uncertainty-guided and failure-aware vision-language-action framework for robotic control. *arXiv preprint arXiv:2604.16677*, 2026.

Jacopo Diquigiovanni, Matteo Fontana, and Simone Vantini. Conformal prediction bands for multivariate functional data. *Journal of Multivariate Analysis*, 189:104879, 2022.

Senyu Fei, Siyin Wang, Junhao Shi, Zihao Dai, Jikun Cai, Pengfang Qian, Li Ji, Xinzhe He, Shiduo Zhang, Zhaoye Fei, et al. Libero-plus: In-depth robustness analysis of vision-language-action models. *arXiv preprint arXiv:2510.13626*, 2025.

Isaac Gibbs and Emmanuel Candes. Adaptive conformal inference under distribution shift. *Advances in Neural Information Processing Systems*, 34:1660–1672, 2021.

Qiao Gu, Yuanliang Ju, Shengxiang Sun, Igor Gilitschenski, Haruki Nishimura, Masha Itkina, and Florian Shkurti. Safe: Multitask failure detection for vision-language-action models. *Advances in Neural Information Processing Systems*, 38:40041–40076, 2026.

Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *arXiv preprint arXiv:1610.02136*, 2016.

Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, et al.

$$\backslash$$

pi_ $\{0.5\}$: $avision - language - actionmodelwithopen - worldgeneralization.$ *arXiv preprint arXiv:2504.16054*, 2025.

Paul F Jaeger, Carsten T Lüth, Lukas Klein, and Till J Bungert. A call to reflect on evaluation practices for failure detection in image classification. *arXiv preprint arXiv:2211.15259*, 2022.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, Quan Vuong, Thomas Kollar, Benjamin Burchfiel, Russ Tedrake, Dorsa Sadigh, Sergey Levine, Percy Liang, and Chelsea Finn. OpenVLA: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024.

Moo Jin Kim, Chelsea Finn, and Percy Liang. Fine-tuning vision-language-action models: Optimizing speed and success. *arXiv preprint arXiv:2502.19645*, 2025.

Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. *Advances in neural information processing systems*, 31, 2018.

Yousung Lee and Dongsoo Har. Perturbation-based uncertainty for failure detection in vision-language-action models. *arXiv preprint arXiv:2606.20754*, 2026.

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

Yaniv Ovadia, Emily Fertig, Jie Ren, Zachary Nado, D. Sculley, Sebastian Nowozin, Joshua V. Dillon, Balaji Lakshminarayanan, and Jasper Snoek. Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

Karl Pertsch, Kyle Stachowicz, Brian Ichter, Danny Driess, Suraj Nair, Quan Vuong, Oier Mees, Chelsea Finn, and Sergey Levine. Fast: Efficient action tokenization for vision-language-action models. *arXiv preprint arXiv:2501.09747*, 2025.

Harshitha Rajaprakash, Aditeya Prajapati, Rong Xue, Abrar Anwar, and Jesse Thomason. Safecast: Robust failure detection for vla policies with contrast-set training and calibration. *arXiv preprint arXiv:2608.04246*, 2026.

Ralf Römer, Adrian Kobras, Luca Worbis, and Angela Schoellig. Failure prediction at runtime for generative robot policies. *Advances in Neural Information Processing Systems*, 38:7631–7670, 2026a.

Ralf Römer, Maximilian Seeliger, Saida Liu, Ben Sturgis, Marco Bagatella, Daniel Marta, Andreas Krause, and Angela P Schoellig. Uncertainty quantification for flow-based vision-language-action models. *arXiv preprint arXiv:2606.18043*, 2026b.

Baochen Sun and Kate Saenko. Deep coral: Correlation alignment for deep domain adaptation. In *European conference on computer vision*, pp. 443–450. Springer, 2016.

Yiyou Sun, Yifei Ming, Xiaojin Zhu, and Yixuan Li. Out-of-distribution detection with deep nearest neighbors. In *International conference on machine learning*, pp. 20827–20840. PMLR, 2022.

Ryan J Tibshirani, Rina Foygel Barber, Emmanuel Candes, and Aaditya Ramdas. Conformal prediction under covariate shift. *Advances in neural information processing systems*, 32, 2019.

Vladimir Vovk, Alexander Gammerman, and Glenn Shafer. *Algorithmic learning in a random world*. Springer, 2005.

Chen Xu, Tony Khuong Nguyen, Emma Dixon, Christopher Rodriguez, Patrick Miller, Robert Lee, Paarth Shah, Rares Ambrus, Haruki Nishimura, and Masha Itkina. Can we detect failures without failure data? uncertainty-aware runtime failure detection for imitation learning policies. *arXiv preprint arXiv:2503.08558*, 2025.

Xueyang Zhou, Yangming Xu, Guiyao Tie, Yongchao Chen, Guowen Zhang, Duanfeng Chu, Pan Zhou, and Lichao Sun. Libero-pro: Towards robust and fair evaluation of vision-language-action models beyond memorization. *arXiv preprint arXiv:2510.03827*, 2025.

A CORPUS DATASHEET AND COLLECTION PROTOCOL

Table 2: **Rollout inventory and analysis scope.** Counts give successes/failures. Only the first two rows enter the primary two-policy analyses, and legacy clean records serve as regression checks. The full index contains 5,501 unique rollouts.

Policy or source	Clean	Perturbed	Role
OpenVLA-OFT	462/38	824/267	primary
π_0	399/101	563/327	primary
$\pi_{0.5}$	460/40	322/54	auxiliary
OpenVLA	248/252	47/97	auxiliary
Legacy π_0	389/111	none	regression only
Legacy π_0-FAST	298/202	none	regression only
Total	3,000	2,501	5,501 rollouts

Inventory scope. The primary analyses use 1,000 clean and 1,981 perturbed rollouts of OpenVLA-OFT and π_0. The auxiliary policies OpenVLA (Kim et al., 2024) and $\pi_{0.5}$ (Intelligence et al., 2025) contribute 1,520 rollouts, and legacy π_0 and π_0-FAST (Pertsch et al., 2025) clean rollouts contribute another 1,000 that serve only as protocol regression tests. Pilot trajectories are already included in the perturbed counts and must not be added again. The two primary perturbed corpora contain 1,387 successes and 594 failures. The dual-class floor is a collection target rather than a claim of equal class proportions, and the realized proportions depend on adaptive allocation and stopping.

Grid and blocked cells. Seven LIBERO-plus dimensions cover the ten LIBERO-10 tasks at severities L1 to L5. Layout-L5 and light-L1 are excluded because source variants are unavailable. Layout-L4 has only eight rollouts over at least three tasks and is treated as exploratory throughout. Each of the 14 policy and dimension combinations covers all ten tasks. Aggregates over the full collected corpus include the exploratory records and should not be read as strictly confirmatory estimates.

Allocation and checks. For each policy and dimension, the budget was $N_d \approx \max(30/\hat{p}_{\text{fail}}, 120)$, capped at 320 and allocated across severities in proportion to $1 - \hat{p}_{\text{succ}} + 0.15$, with both rates estimated from a pilot of 256 rollouts per policy. Each variant has a single initial state and contributed at most two episodes, and collection stopped early once both classes reached 30 with $N \geq 120$. Live checks compared observed success rates with composition-adjusted anchors using Jeffreys-smoothed cell rates. They were evaluated only at $n \geq 40$ with a one-sided binomial test at $p < 0.05$, and deviations on the upper side were only logged. Two lower-side events, π_0 camera and π_0 language, were adjudicated as pilot-anchor optimism combined with variant heterogeneity, since per-cell Fisher tests were not significant after Holm correction. One non-monotonicity, in which π_0 language-L1 was harder than L2, was audited at the instruction level. All eight instructions were valid and the failures were motor failures, so it is recorded as evidence of variant heterogeneity. Checks for NaN values were clean throughout, and every rollout has a side-by-side 512×256 video.

Table 3: **Per-dimension composition of the primary perturbed corpus.** Entries give successes/failures. Marginal top-ups to reach the 30-failure floor (OFT layout $+8$, OFT light $+13$, and π_0 layout $+8$) used empirical rates from the main corpus and are documented separately.

Dimension	OpenVLA-OFT	π_0
Background texture	142/18	78/42
Camera viewpoint	70/50	55/65
Language instruction	149/11	86/54
Object layout	128/30	98/30
Lighting	223/30	98/34
Sensor noise	74/46	97/33
Robot initial state	38/82	51/69
Total	824/267	563/327

Clean corpora. Clean rollouts were collected on the same server, client, and checkpoint stack as the perturbed rollouts, with 50 episodes for each of the ten tasks under the official rotation of initial states. OFT has $462/38$ and π_0 $399/101$ successes/failures, both within 10 percentage points of their reference success rates. Author-released rollouts serve only as protocol regression tests and are excluded from all scientific analyses.

B METRICS, CALIBRATION, AND STATISTICS

Score construction. Feature sequences are truncated to the minimum length within each task. Let h_t be a standardized feature, and let d_+ and d_- denote distances to the success and failure reference distributions. The implemented Mahalanobis and kNN scores are

$$s_t = \sum_{u=1}^{t} \big(d_+(h_u) - d_-(h_u)\big). \tag{4}$$

Mahalanobis uses separate reference means and regularized covariance matrices for the two classes. The kNN variants average the ten nearest distances within each outcome class. Reference steps are capped deterministically at 12,000 per class when necessary. These implementations are therefore supervised, signed distance contrasts rather than success-only novelty scores. SAFE-MLP accumulates sigmoid outputs, whereas SAFE-LSTM emits sigmoid outputs without cumulative summation. LogpZO is evaluated through its reference model interface without an additional cumulative operation in the collection wrapper, and differences of its cached scores alone do not establish that its internal score is additive.

Implemented calibration rule. At timestep t, let n_t be the number of calibration-success trajectories still present. Our calibration routine computes

$$q_t = \min\left\{ \frac{\lceil (n_t+1)(1-\alpha) \rceil}{n_t}, 1 \right\}, \qquad \delta_t = Q_{q_t}^{\text{higher}}\{s_{i,t}\}_{i=1}^{n_t}, \tag{5}$$

where Q^{higher} is the empirical quantile with higher interpolation. A rollout alarms if any score exceeds its band. The task-transfer and repair code extends the last band value when an evaluation trajectory outlasts calibration support. Because quantiles are computed separately at each timestep and clipped to observed support, this is not a simultaneous functional conformal band, and pointwise coverage cannot stand in for the probability of no crossing over the entire trajectory. The reported quantities are therefore empirical FPR, coverage $1 - \mathrm{FPR}$, and the change in FPR relative to clean evaluation. Certifying trajectory-level coverage for the functional calibration intended by SAFE would require evaluating that construction separately.

Ranking and operating points. The primary shift and repair analyses rank rollouts by $\max_t s_t$, whereas the task-factorial code uses the terminal score s_T. The two coincide for the nondecreasing MLP score but need not coincide for the LSTM. Architecture comparisons in Appendix F follow the factorial convention. The operating grid is $\alpha \in \{0.05, 0.1, 0.15, 0.2, 0.3, 0.5\}$, with 0.1 and 0.2 used for the principal comparisons. A negative $\Delta\mathrm{FPR}$ measures fewer alarms on successes and by itself says nothing about alarms on failures, which is why we always read it together with $\mathrm{SpecAUC}$.

Uncertainty and repeated observations. Per-seed primary metrics use $1{,}000$ trajectory-bootstrap replicates. The summaries in Figure 2 instead report bootstrap intervals over the five seed-level estimates, and the degradation summaries likewise bootstrap seed estimates. Paired Wilcoxon comparisons use the five seeds with the recorded multiple-comparison corrections. These intervals describe variability under the stated resampling unit. They do not capture every source of uncertainty, and repeated trajectories do not create independent evidence. The factorial uses $10{,}000$ split-bootstrap replicates conditional on the available ten-task corpus. Feature-drift correlations use 160 OFT and 165 π_0 seed and cell observations, while Figure 7b displays severity-cell means. Their intervals do not account for all dependence induced by trajectories that are shared across seeds.

Feature aggregation. Selection among `First`, `First&Last`, `Last`, and `Mean` uses clean cross-validation within the training data. Table 4 reports the operating point of each aggregation on the

two strong shifts. Across aggregations, OFT $\mathrm{SpecAUC}$ spans 0.835 to 0.875 on camera and 0.851 to 0.869 on robot initialization, with $\Delta\mathrm{FPR}$ between 0.148 and 0.175 and between 0.204 and 0.262. For π_0 the $\mathrm{SpecAUC}$ ranges are 0.762 to 0.825 and 0.778 to 0.806, and the $\Delta\mathrm{FPR}$ ranges are 0.001 to 0.078 and 0.079 to 0.255. The OFT conclusions therefore do not hinge on the aggregation, whereas the size of the π_0 effects does.

Table 4: **Sensitivity to feature aggregation on the two strong shifts.** Each entry is $\mathrm{SpecAUC}/\Delta\mathrm{FPR}$ at $\alpha = 0.1$.

Policy	Aggregation	Camera	Robot-init
OFT	First	0.850/0.163	0.851/0.262
OFT	First&Last	0.875/0.148	0.867/0.214
OFT	Last	0.852/0.175	0.868/0.208
OFT	Mean	0.835/0.162	0.869/0.204
π_0	First	0.819/0.017	0.785/0.107
π_0	First&Last	0.825/0.078	0.787/0.255
π_0	Last	0.762/0.001	0.778/0.079
π_0	Mean	0.800/0.045	0.806/0.167

C Full per-dimension and severity-resolved results

Table 5 reports $\Delta\mathrm{FPR}$ at $\alpha = 0.1$ and 0.2, $\mathrm{SpecAUC}$, and $\mathrm{ShiftResp}$ for SAFE-MLP. The complete long table over detectors, dimensions, and six values of α, with 504 rows and confidence intervals, will be included in the code release; Table 5 reports the rows used in the main text.

Table 5: **SAFE-MLP per-dimension results.** Values are means over five seeds.

Policy	Dimension	$\Delta\mathrm{FPR}(0.1)$	$\Delta\mathrm{FPR}(0.2)$	$\mathrm{SpecAUC}$	$\mathrm{ShiftResp}$
OFT	background	$+0.114$	$+0.179$	0.750	0.633
OFT	camera	$+0.175$	$+0.313$	0.852	0.703
OFT	language	-0.007	-0.033	0.789	0.490
OFT	layout	$+0.040$	$+0.141$	0.726	0.620
OFT	light	$+0.077$	$+0.146$	0.823	0.621
OFT	noise	$+0.119$	$+0.167$	0.843	0.610
OFT	robot-init	$+0.208$	$+0.218$	0.868	0.653
π_0	background	-0.050	-0.046	0.820	0.492
π_0	camera	$+0.078$	$+0.049$	0.825	0.557
π_0	language	$+0.113$	$+0.067$	0.767	0.566
π_0	layout	$+0.081$	$+0.070$	0.840	0.559
π_0	light	$+0.016$	$+0.027$	0.906	0.567
π_0	noise	$+0.002$	$+0.027$	0.793	0.564
π_0	robot-init	$+0.255$	$+0.218$	0.787	0.661

Severity-resolved degradation. Figure 7 resolves SAFE-MLP AUROC and successful-rollout coverage by policy, dimension, and severity, with coverage defined as $1 - \mathrm{FPR}$ at $\alpha = 0.1$. Several cells keep strong ranking despite low coverage. OFT robot-init L5, for example, has an AUROC of 0.93 and a coverage of 0.20, although its small class count limits precision. Grey AUROC cells include cells with insufficient failure support as well as unsampled conditions. Severity trends are not uniformly monotone, so a pooled benchmark score conceals both heterogeneity and sampling limits.

Feature drift and false-alarm inflation. Hidden-state displacement is associated with threshold drift. Across seed and severity-cell observations, median Mahalanobis feature drift correlates with MLP false-alarm inflation at Spearman $\rho = 0.64$ for OFT and 0.38 for π_0. This is a descriptive association rather than a validated predictor for new shifts, and repeated observations share trajectories across seeds, which limits the independence behind the reported intervals. Language-length associations and score-prefix interventions in Appendix D further separate sensitivity to initial observations from accumulated execution evidence.

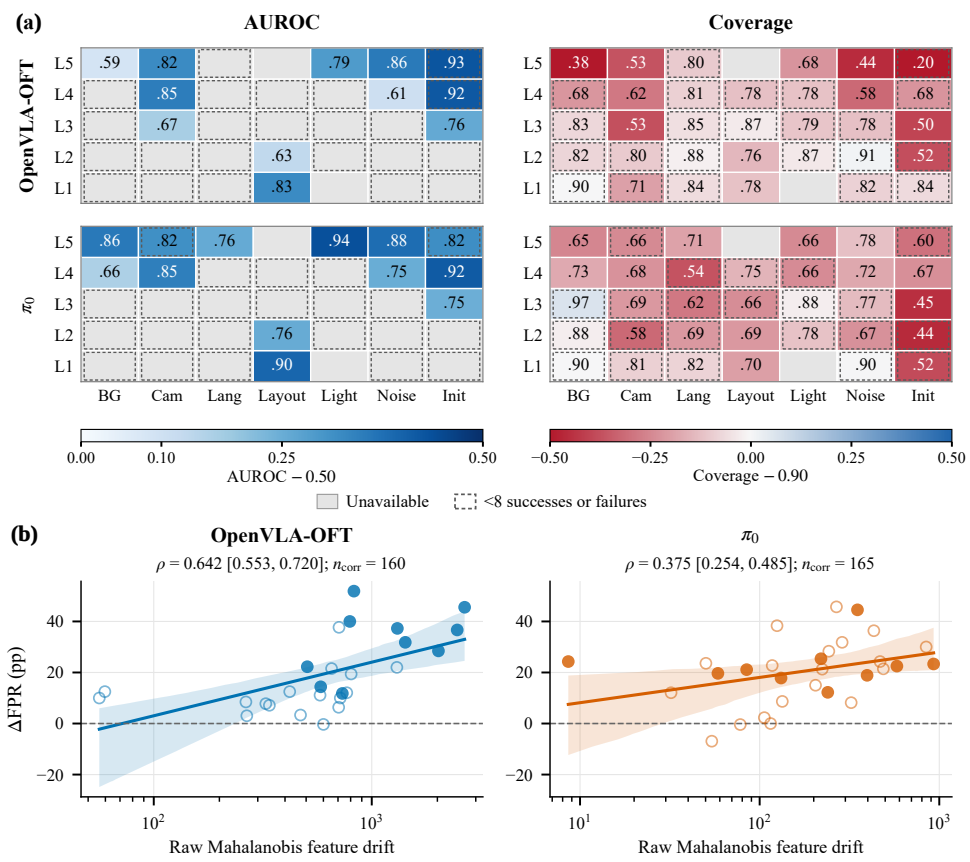

Figure 7: **SAFE-MLP degradation varies across conditions and severities.** Panel a prints raw AUROC and coverage at nominal $1-\alpha = 0.90$, with color encoding AUROC above 0.50 or coverage relative to 0.90. Grey cells are unavailable or lack sufficient support for AUROC, and dashed borders mark fewer than eight successes or failures. Panel b plots severity-cell false-alarm inflation against feature drift on a logarithmic horizontal axis. Filled points have at least eight outcomes of each class, and the displayed correlations use the seed-level observations described in Appendix B.

D TEMPORAL MECHANISM AND ALARM TIMING

Score and geometry checks. The audit checks distance-score signs, aggregation, and truncation against the reference pipeline. Synthetic mean shifts of 0.5, 1.0, and 2.0 trajectory-time means injected into clean evaluation features increase ΔFPR for Mahalanobis and for both kNN variants. Real camera and robot-initialization perturbations instead move the signed Mahalanobis increments downward. The class-relative distance difference explains why distance from the clean mean alone does not fix the direction of alarms. This evidence concerns score behavior and does not establish that our pointwise calibration is equivalent to the functional calibration of SAFE.

Temporal controls. For OFT, the audited increment medians shift by approximately -800 score units on camera and $-1{,}200$ on robot initialization. The maximum band margin $\max_t(s_t - \delta_t)$ becomes more negative in all 20 policy, dimension, and seed comparisons (Figure 8a). The median gap below the band grows by about 894 and 1,336 score units for OFT and by 41.0 and 49.4 for π_0. Successful trajectories average 27.34 feature steps clean and 27.17 perturbed in the OFT comparison (Figure 8b). These lengths weaken a simple duration account without excluding all censoring or task-composition effects. Clean FPR spans 6% to 22% at nominal $\alpha = 0.1$ across the audited configurations, so some settings already exceed nominal before any condition shift. The downward response of LogpZO is reported separately from this Mahalanobis mechanism.

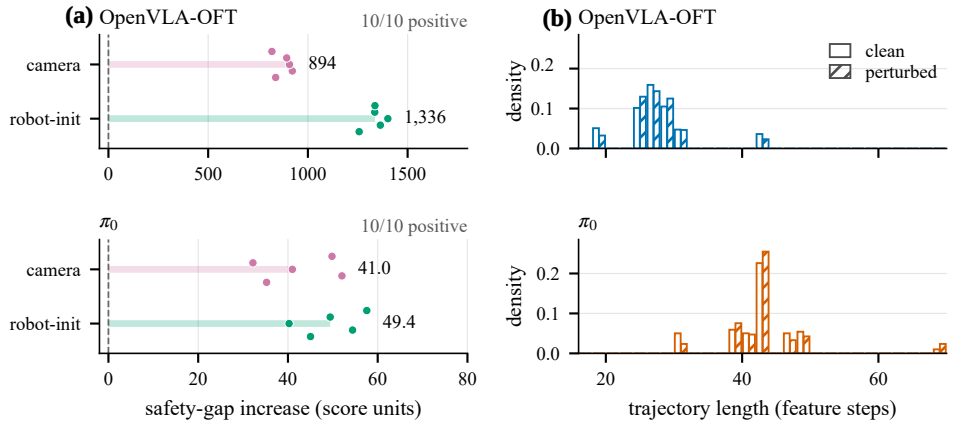

Figure 8: **Additional controls for Figure 3.** Panel c shows positive changes in the median gap below the band for all 20 Mahalanobis comparisons. Dots are seeds, printed values are medians, and the two policies use separate horizontal scales. Panel d compares clean outline histograms with hatched perturbed histograms of successful-trajectory lengths. Similar lengths do not explain the direction of score displacement.

Score-prefix interventions. For the additive MLP, replacing a prefix with same-task clean increments is an exact intervention on the cached score sum. It does not produce the hidden states or the behavior that a robot would generate under a physical intervention. At a splice point of $0.25T$, the median first-crossing displacement for OFT is 7.75, 10.0, and 7.5 feature steps (145.5, 126.3, and 144.2 environment steps) for camera, light, and robot-init, and the splice removes 30.7%, 34.7%, and 18.7% of alarms. The initial-observation comparison accounts for 28.7%, 37.0%, and 27.8%. For π_0, crossing displacements are 0.25, 0, and 1.0 feature steps, with initial-observation shares of 0.6%, 0%, and 2.3%. The score shift at $t = 0$ has a Cliff's δ of 0.217, 0.237, and 0.240 for OFT on camera, noise, and robot-init, while π_0 reaches 0.174 on robot-init. The splice grid at $0.1T$ and the per-task results will be included in the code release. LSTM prefix splicing is excluded because its recurrent state does not admit this additive replacement.

Language associations. The analysis finds a positive association between instruction length and π_0 score displacement, while OFT stays near its language-control level. In the collected corpus, OFT failures make up 6.9% of language-perturbed rollouts versus 7.6% of clean rollouts, and the corresponding π_0 fractions are 38.6% and 20.2%. These comparisons are descriptive because adaptive allocation changes the sampled mixture of variants. They support different sensitivity profiles without identifying instruction neglect as the only possible explanation.

E VISUAL CASE STUDIES AND SCORE-PREFIX REPLAY

Sources, selection, and temporal scope. Figures 4, 9, and 10 reuse the original rollout videos, the indexed outcome labels, and the cached scores of the main analysis. We fix seed 0, $\alpha = 0.1$, and the clean-selected feature aggregations. Bands are reconstructed from the clean calibration successes of that seed only, 116 for OFT and 100 for π_0 at $t = 0$, with the original pointwise calibration function of Equation 5 and without retraining or threshold tuning. Clean examples and prefix donors are held-out evaluation rollouts. The cases were selected after inspecting outcomes and alarm patterns and were then checked visually, to illustrate distinct decisions on interpretable tasks. They are not a random sample, a new significance test, or a causal comparison with matched initial states. Case-level values, rollout identifiers, plotted scores, and frame indices are recorded in separate case-study manifests, and the aggregate results are unchanged.

For these plots, the zero-based feature index t maps to the environment step e through the stored `env_step_index` array rather than by stretching a truncated score trajectory to the duration of the video. The video writer records the initial observation and every second environment step. Image connectors in Figure 9 use exact matches between recorded and scored steps. Elsewhere image labels

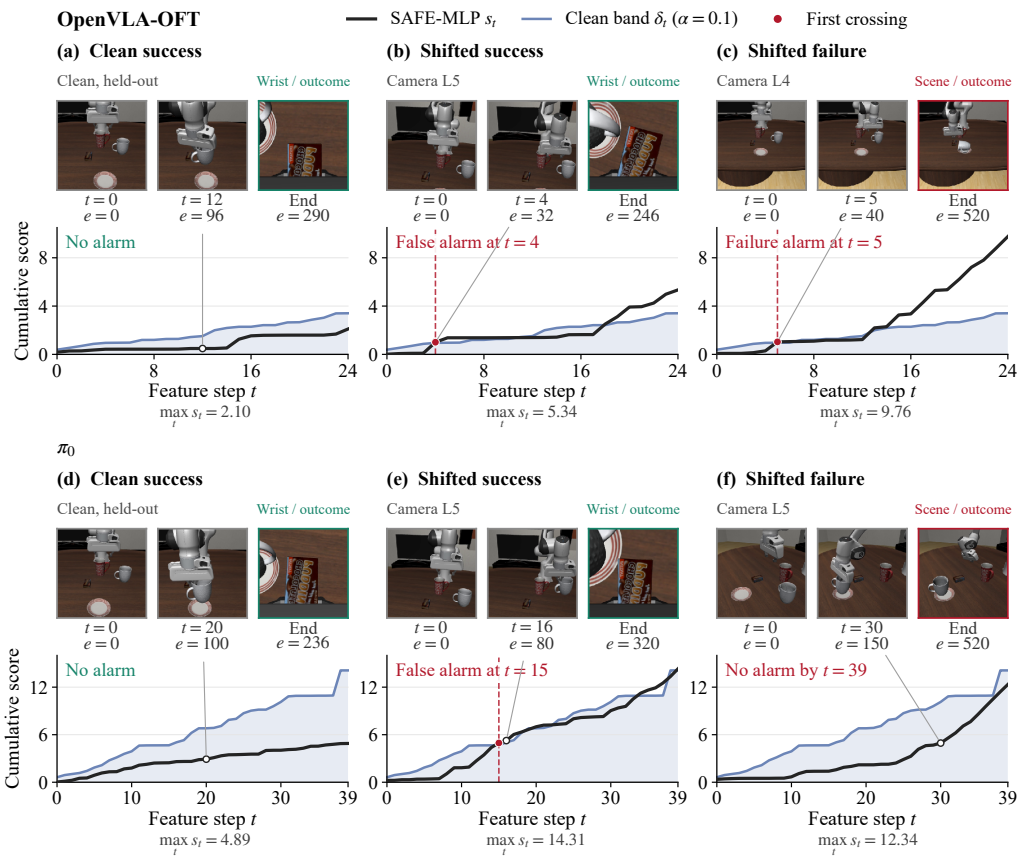

Figure 9: **Robot outcomes and monitor decisions disagree under camera shift.** Six recorded rollouts of one task are scored by the SAFE-MLP of seed 0 at $\alpha = 0.1$. Black curves are cumulative scores, and the blue boundary is the band calibrated on clean successes, with shared axes within each policy row. Red dots mark first crossings. Panels a and d are clean successes without alarms, b and e succeed despite alarms, c is a detected failure, and f is a failure without an alarm in the scoring window. The first two images of each panel are agent-camera observations at the indicated feature and environment steps (t, e), and the last image shows the recorded outcome after the scoring window. Scores stop at $t = 24$ ($e = 192$) for OFT and at $t = 39$ ($e = 195$) for π_0 under the original truncation protocol, and alarm time is not a failure-onset annotation.

give the actual recorded step, so that, for example, $e = 224$ is the last recorded observation before a score at $e = 225$. Terminal frames establish the recorded outcome and can lie outside the evaluation window. Success and failure labels are episode-level environment outcomes, and we introduce no manual labels of failure onset. All images are decoded original observations from the identified camera, and all alarms are evaluated offline on the recorded executions.

Different detectors on the same failed execution. Figure 4 in Section 5 makes the operational consequence of score construction visible on two further tasks. On the OFT caddy-placement failure under robot-init L3, SAFE-MLP first alarms at $t = 5$ ($e = 40$) and L2-kNN at $t = 1$ ($e = 8$), whereas Mahalanobis stays below its band through $t = 18$ ($e = 144$). On the π_0 microwave failure under robot-init L5, the corresponding crossings are at $t = 13$ ($e = 65$) and $t = 11$ ($e = 55$), with no Mahalanobis crossing through $t = 45$ ($e = 225$). Both episodes end unsuccessfully at $e = 520$. The signed margin $m_t = s_t - \delta_t$ exposes these disagreements directly. A negative margin denotes the absence of an alarm rather than evidence of successful execution, and each detector keeps its own score units and clean-calibrated band. These examples complement the score-displacement analysis of Section 5 and do not by themselves identify the causal contribution of the shift or the onset of physical failure.

What the early score prefix contributes. Figure 10 revisits the two false alarms of Figure 9b and e with the replay rule of Appendix D. Write $u_t = s_t - s_{t-1}$ with $s_{-1} = 0$, and let u_t^{clean} be the increments of the same-task clean donor. With $k = \lceil T/4 \rceil$, only the first k increments are replaced,

$$\widetilde{s}_t = \begin{cases} \sum_{j=0}^{t} u_j^{\text{clean}}, & t < k, \\ \sum_{j=0}^{k-1} u_j^{\text{clean}} + \sum_{j=k}^{t} u_j, & t \geq k. \end{cases} \tag{6}$$

Donors follow the original deterministic SHA-256 assignment within the same-task clean evaluation successes of the seed, and neither the donor nor the prefix length is optimized for these figures. For OFT, replacing $k = 7$ increments moves the first crossing from $t = 4$ to $t = 20$ ($e = 32$ to $e = 160$). For π_0, replacing $k = 10$ increments moves it from $t = 15$ to $t = 14$ ($e = 75$ to $e = 70$). The OFT example thus depends strongly on its early score offset, while this π_0 example still alarms near the same point. The population-level comparison remains the prefix analysis of Appendix D rather than these two selected examples. The replay is a score-level intervention with a fixed band and unchanged later increments. It is not a new robot rollout, and it makes no claim that substituting observations would preserve the actions of the policy. The images show only inputs and outcomes that were actually recorded.

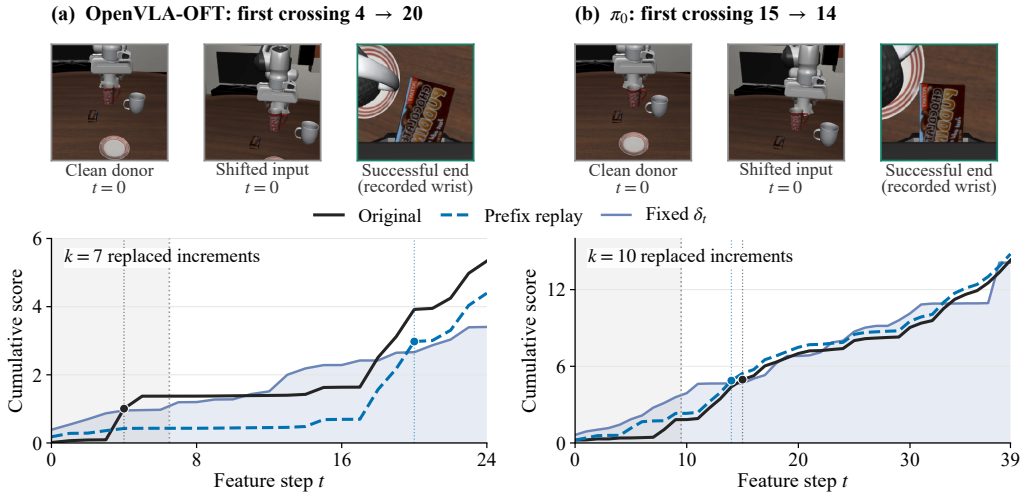

Figure 10: **Two false alarms respond differently to replacing their early score prefixes.** Original observations show the clean donor, the camera-shifted input, and the actual successful terminal wrist view. Black curves are the original SAFE-MLP scores, and dashed blue curves replace only the first quarter of increments with those of the assigned clean donor. The calibration band is fixed, grey shading marks the replaced prefix, and dots mark first crossings. The OFT crossing moves 16 feature steps later and the π_0 crossing one step earlier. The replay changes scores only, and the terminal images are not counterfactual execution results.

F Additional Task and Condition Analyses

Interpretation of the main effects. The AUROC loss along the task axis exceeds the loss along the condition axis in the exhaustive MLP analysis, and this need not extend to other probe architectures. On the five representative splits, the OFT LSTM reaches a B-cell AUROC of 0.832 while the MLP reaches 0.460. The representative splits were selected with the interaction statistic, so they do not estimate an unconditional architecture advantage over all splits. The failure-count intervention changes the B-cell AUROC by $+2.53$ points between 15 failures and the full set, with an interval that crosses zero. It therefore gives no clear evidence that adding failures within the tested range resolves task transfer, but it does not rule out benefits outside that range or from different failure examples. Different failure compositions in cells B, C, and D also complicate a causal comparison of the task and condition axes.

Design. Each split assigns seven seen and three unseen tasks, and the seen-task clean data are split 55/25/20 into training, calibration, and evaluation with stratification. The exhaustive layer covers

all 120 splits and five seeds for SAFE-MLP, which gives 1,200 trainings with cached scores. A representative layer of five splits, selected at the minimum, first quartile, median, third quartile, and maximum of the pooled I, additionally runs the LSTM with saved weights. All interaction intervals use 10,000 split-bootstrap replicates, and the logit-transformed I uses a finite-sample pair correction at the boundaries. Logit-transformed interactions are 1.59 for OFT and 0.91 for π_0, and the Spearman associations between I and $C - A$ have $p = 1.3 \times 10^{-7}$ and $p = 5.9 \times 10^{-5}$. On the coverage axis the interaction is $+2.2$ and $+2.7$ points for π_0 and mixed near zero for OFT.

Representative-layer caveat. Selection on I correlates with extreme values of B, so marginal cells are reported from the exhaustive layer only. On the representative splits, the OFT MLP reaches $\mathrm{B} = 0.4597\ [0.3798, 0.5448]$, which is statistically indistinguishable from chance, while the OFT LSTM reaches 0.832, a gap of $+0.3725\ [0.2956, 0.4516]$.

Failure-count dose response. We subsample π_0 seen-task training failures to 15, 25, 40, and the full set (44.9 on average), with 10 stratified splits and five seeds per dose, for 200 trainings. The B-cell AUROC moves from 0.6095 to 0.6214, 0.6362, and 0.6348, and the gain from 15 failures to the full set is $+2.53$ points $[-0.29, 5.77]$ (Figure 11). Interpretation is restricted to this intervention. Comparisons are internal to the split set of this ablation, whose full-data mean differs from the exhaustive layer through subset selection. The tested range is 15 to 45 failures, so deeper starvation is untested, and cells A were not re-measured under subsampling, so comparisons use absolute B values. The observational within-split correlation, with $\rho = 0.17$ and $p = 0.061$ for OFT and $\rho = -0.12$ for π_0, is confounded by task composition, and the controlled subsampling is the more direct comparison.

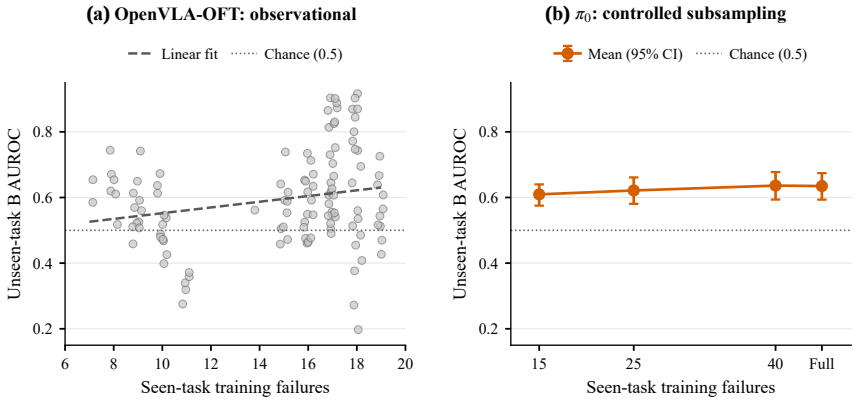

Figure 11: **Failure-count dose response on a shared AUROC scale.** Panel a shows the observational split-level B-cell AUROC of OpenVLA-OFT against seen-task training failures, where the dashed trend ($\rho = 0.17$) is not significant and is confounded by task composition. Panel b shows controlled π_0 subsampling over 10 stratified splits and five seeds per dose, with means and 95% confidence intervals. The estimate of $+2.53$ points from 15 failures to the full set has an interval that crosses zero.

Table 6: **Deployment-proxy cell D from the exhaustive 120-split layer.** Intervals are split-bootstrap 95% intervals conditional on the observed task corpus.

Policy	α	AUROC	FPR	Coverage
OFT	0.1	0.743 [0.727,0.758]	0.508 [0.474,0.544]	0.492 [0.457,0.528]
OFT	0.2	0.743 [0.728,0.758]	0.780 [0.755,0.804]	0.220 [0.196,0.245]
π_0	0.1	0.693 [0.680,0.706]	0.408 [0.364,0.447]	0.592 [0.553,0.633]
π_0	0.2	0.693 [0.680,0.707]	0.598 [0.558,0.636]	0.402 [0.367,0.439]

Split sensitivity. Across all 120 splits, the B-cell AUROC has a population standard deviation of 0.145, an interquartile range of 0.161, and a range of $[0.197, 0.916]$ for OFT, and 0.107, 0.134, and $[0.349, 0.897]$ for π_0. Resampling only three splits yields 95% ranges of the mean estimate of

$[0.434, 0.761]$ and $[0.543, 0.781]$. For comparison, SAFE reports split and seed standard deviations of 5 to 15 points in Table 8 of Gu et al. (2026).

G ADDITIONAL REPAIR ANALYSES

Scope and information budgets. The pool consists of 155 language-L1 and layout-L1 rollouts, and the remaining perturbed evaluation set contains 1,826 rollouts with no overlap. Repair summaries average all evaluated dimensions, not only camera and robot initialization. Static calibration methods use five representative task splits and five seeds, while the feedback rule and the three-family comparison use the median representative split with five seeds. The plotted summaries therefore do not form a fully paired comparison across every method.

Quantile inflation. The method labeled Robust CP replaces α in Equation 5 with $\max(\alpha - \rho, 10^{-6})$ for $\rho \in \{0.05, 0.1, 0.2\}$. It is inspired by conservative conformal calibration (Cauchois et al., 2024), but the evaluated routine neither solves an f-divergence robust-quantile problem nor verifies a divergence bound. All 100 audited selected configurations equal the per-timestep maximum of the clean calibration scores. At these settings, the mean absolute coverage error is 7.18 and 9.90 points for OFT at $\alpha = 0.1$ and 0.2 and 9.20 and 12.05 points for π_0, with a TPR of about 35.9% and 35.7%. Absolute coverage error differs from the fraction of successes that cross the band, and the same band can have different absolute errors under different nominal targets. A band above the empirical maximum lies outside the tested family, so these results do not establish a universal lower bound for static thresholds.

Feedback adaptation. The method labeled ACI uses step sizes $\gamma \in \{0.005, 0.01, 0.02\}$, two arrival orders, and five stream permutations. Its implemented update is

$$\alpha_{k+1} = \mathrm{clip}_{[0.001, 0.5]}\big(\alpha_k + \gamma(\alpha - \mathbf{1}\{Y_k = \text{success}, A_k = 1\})\big), \tag{7}$$

where A_k indicates an alarm and Y_k is the observed rollout outcome. The update is applied after every rollout, failures included, and assumes immediate outcome feedback. Its error indicator therefore concerns the joint event of success and alarm rather than the success-conditional FPR used in evaluation, and clipping and the finite empirical band impose further restrictions. We report it as a rule inspired by ACI (Gibbs & Candes, 2021) and do not claim to evaluate the original guarantee for this conditional objective. A success-only update and delayed feedback require separate experiments.

Re-anchoring and weighting. Re-anchoring estimates $\widehat{\Delta}$ from the difference between the increment medians of the pooled deployment rollouts and of the clean calibration rollouts, and then adds $t\widehat{\Delta}$ to the band. The estimator that uses only pool successes serves as a diagnostic oracle and requires labels. The cross-cell criterion requires an oracle-relative bias below 50% on at least two dimensions. Robustified variants, which we call c-robust, keep the lowest 50%, 60%, or 70% of trajectories by maximum band margin and use trimmed estimators, and no pool cell satisfies their criterion (Table 7). Estimated weighting uses a gradient-boosted classifier on six score-trajectory summaries, namely three quantiles, the slope, the terminal score, and the band-crossing count, with weights clipped to $[0.05, 20]$. These are ratios estimated from a pool with mixed outcomes rather than known success-conditional density ratios. The criterion requires coverage within three points of nominal on at least five of seven dimensions and a TPR loss of at most five points at matched coverage, and no dimension passes for either policy.

Data and representation interventions. SAFECAST-style TrainAug freezes the first MLP layer and retrains its head with labeled pool rollouts, CalAug adds pool successes to calibration, and the combined variant does both. These in-suite adaptations are not full reproductions of SAFECAST. CORAL whitens and recolors the saved 256-dimensional pre-ReLU MLP features using the unlabeled pool and then re-scores them with the saved head. Recurrent probes are excluded from this intervention. The data-side variants improve the measured trade-off for some π_0 settings without uniformly improving OFT.

Configuration selection and interpretation. For each policy and α, the analysis selects the configuration with the smallest evaluation coverage error among the swept settings. Such best-of-sweep readouts are retrospective diagnostics rather than hyperparameters chosen without deployment

labels, and they can overstate what an unlabeled deployment procedure would achieve. The empirical envelopes also combine methods with different information budgets and split coverage, so they should be read as a descriptive comparison of tested operating points.

Table 7: **Auxiliary-pool contamination and label-free re-anchoring bias.** Bias is relative to the success-only oracle, and the c-robust variants pass none of the four cells.

Policy	Pool cell	Failure share	Relative bias	c-robust criterion
OFT	language-L1	6.2%	48%	fail
OFT	layout-L1	16.4%	60%	fail
π_0	layout-L1	27.5%	317%	fail
π_0	language-L1	31.2%	156%	fail

Table 8: **Recorded repair criteria.**

Method	Criterion result	Key check
Quantile inflation	family saturation	$100/100$ selected bands equal clean maxima
Feedback rule	observed trade-off	dependent on arrival order and γ
Naive re-anchoring	contaminated	four-cell bias of 48% to 317%
c-robust re-anchoring	$0/4$ cells pass	pseudo-success and trimmed variants
Weighted FCP	$0/7$ dimensions pass per policy	success-only weights as diagnostic
TrainAug and CalAug	empirical comparison	head-only in-suite instance
CORAL	empirical comparison	MLP only, 256-d pre-ReLU features

H Reproducibility details

Environments and checkpoints. The policy environment uses `torch` 2.2.0+cu121, `torchvision` 0.17.0, `timm` 0.9.10, `tokenizers` 0.19.1, and the OpenVLA-OFT bidirectional-attention `transformers` fork (commit `bc339d9a`), matching the upstream `pyproject` pins. The simulation environment uses `robosuite` 1.4.x and `mujoco` 2.3.7 with LIBERO, and the analysis environment uses the `pyarrow` 17.0.0 stack. Revision-pinned checkpoints include the OpenVLA-OFT LIBERO-10 finetune (`95220f9a`), the official π_0 `pi0_libero` checkpoint (with per-file hashes recorded), OpenVLA (`8097032`), and the official $\pi_{0.5}$ `pi05_libero` checkpoint. During collection, the success-rate checks detected an environment misconfiguration in which a policy environment had been assembled without the required attention fork, which silently degrades parallel decoding. The affected rollouts were discarded and re-collected.

Compute. Collection GPU time was not logged separately, so we do not estimate it. Analysis, including probe training for the 1,200 factorial runs and the 200 dose-response runs, used fewer than 30 GPU-hours, and the repair study reused cached scores with essentially no additional GPU time.

Seeds and ledger. We use five seeds throughout and verified the enumeration of the $\binom{10}{3} = 120$ splits, with no gaps or duplicates. Every reported number is indexed in a 545-entry provenance ledger that links it to its artifact path and verification record; the ledger will be released with the code.

I Discussion and limitations

Implications for evaluation and deployment. A deployment-facing evaluation of a failure detector should report, for each perturbation dimension, failure discrimination under shift, shift response among successes, and the change in false alarms, together with the clean operating point. The last matters because clean false-alarm rates already range from 6% to 22% at nominal $\alpha = 0.1$ across our audited configurations. Neither a high AUROC nor a falling false-alarm rate certifies a monitor, and any unlabeled adaptation should state how it handles failures in its calibration pool. Our results also indicate where effort should go. Scores should be built so that per-step evidence is insensitive to benign change, because accumulation amplifies any bias, and calibration under shift needs labeled shifted successes rather than unlabeled pools.

Limitations. Our evidence comes from one simulated benchmark family, two primary policies, and a class-targeted collection. Simulation makes controlled and repeatable shifts possible, but real-world shifts are richer and can occur together. Several severity cells have limited outcome support, and repeated seeds and task splits reuse trajectories, so our intervals do not capture all dependence. The implemented band is pointwise and does not certify simultaneous trajectory coverage, which is why we report measured operating points rather than violated guarantees. The repairs are adaptations of the cited methods, evaluated with cached features and a limited pool (Appendix G). Score-prefix replay intervenes on cached scores rather than on robot execution, and we evaluate alarms without executing a recovery controller, so alarm quality does not directly measure reduced physical risk.