CALIBRATED MONITORING OF INSTRUCTION–EXECUTION MISMATCH IN VISION-LANGUAGE-ACTION POLICIES

Anonymous authors
Paper under double-blind review

ABSTRACT

Vision–language–action (VLA) policies can execute familiar motions while failing to follow the current instruction. Such instruction–execution mismatch can provide weak signals for monitors based on action uncertainty or novelty. To address this, we introduce CALM, a **CAL**ibreated **M**onitor that decodes instruction-conditioned spatial support from frozen VLA representations. Specifically, CALM construct a Bayesian last layer (BLL) model to predict a distribution over end-effector locations together with an abstention output. After scalable variational training on successful execution endpoints and absent-target examples, we accumulate spatial support deficits from dispersed or abstaining predictions based on the trained BLL model at runtime and request assistance when the cumulative score exceeds a calibrated threshold. To obtain this threshold, we rely on the split conformal prediction (CP) which entails held-out calibration episodes. Under episode-level exchangeability, split CP provides a finite-sample bound on the marginal probability of interrupting an otherwise successful episode. On LIBERO-10 with $\pi_{0.5}$, π_0, and OpenVLA, CALM achieves the highest detection rate, balanced accuracy, and time-weighted accuracy among evaluated methods within the empirical false-positive-rate budget.

1 INTRODUCTION

Vision–language–action (VLA) policies combine large multimodal models with expressive action generators for general-purpose robot manipulation (Chi et al., 2023; Lipman et al., 2023; Black et al., 2025; Physical Intelligence et al., 2025). Reliable deployment, however, also requires deciding when autonomous execution should continue and when assistance is needed (Agia et al., 2025; Ren et al., 2023; Yuan et al., 2026). This decision is difficult in familiar scenes, where VLAs may continue learned visuomotor routines after the language instruction changes (Fang et al., 2026; Zhou et al., 2025; Fei et al., 2026). Such execution can remain mechanically coherent and action-confident while violating the current instruction. We study this failure mode as *instruction–execution mismatch*.

To address such mismatch, one line of works develops runtime monitors that can primarily detect failures through action uncertainty, temporal consistency, representation novelty, or learned failure likelihood (Agia et al., 2025; Römer et al., 2025; Gu et al., 2025; Huang et al., 2026; Rao et al., 2026; Römer et al., 2026; Seligmann et al., 2026). These signals can respond weakly when the physical behavior remains familiar despite a changed instruction (Fang et al., 2026; Zhou et al., 2025; Fei et al., 2026). Figure 1 illustrates this case: the VLA continues to manipulate familiar objects, while *Flow Instability*, our proxy for action-generation uncertainty (Rao et al., 2026), changes little. In contrast, its attention maps respond to the instruction, suggesting that instruction-sensitive information remains available in the VLA's vision–language representations.

Alternatively, existing alignment approaches exploit such semantic information using separate instruction–action verifiers (Kwok et al., 2026), external VLMs that judge observed behavior (Grislain et al., 2026), or predicted-outcome verification (Wu et al., 2025; 2026). These approaches introduce additional inference, while VLM-based verification can face a representational gap in interpreting low-level action consequences (Wu et al., 2025). Outcome-based verification additionally depends on the accuracy of world-model or simulation predictions (Wu et al., 2025; 2026). This motivates extracting instruction-sensitive evidence directly from the frozen VLA.

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.

Indeed, frozen VLA representations contain spatial and task-relevant structure accessible through lightweight readouts (Lu et al., 2025; Zhang et al., 2026b; Gu et al., 2025). ProbeAct, for example, decodes object positions from internal VLA features, but combines them with robot–object kinematics to detect grasping and placement failures (Zhang et al., 2026a). Its intervention signal therefore remains tied to physical execution abnormalities, which need not arise during smooth execution of the wrong instruction. The remaining challenge is to make instruction-conditioned spatial support itself informative for intervention.

To address this gap, we introduce CALM, a **CAL**ibrated **M**onitor of instruction–execution mismatch in VLAs. CALM learns a Bayesian spatial readout from frozen VLA representations using successful execution endpoints and absent-target abstention examples. A variational Bayesian last layer produces a posterior predictive distribution over spatial execution locations and abstention (Harrison et al., 2024), from which we define a spatial support deficit that increases as the predictive distribution becomes more spatially dispersed. Thus, the monitor converts instruction-sensitive internal representations into physically grounded evidence without an external semantic verifier.

To further convert this evidence into reliable assistance decisions, we will adapt the conformal prediction (CP) framework. Previous CP-based methods typically support calibrated assistance and abstention (Ren et al., 2023; Yuan et al., 2026; Singh et al., 2026), as well as trajectory-level false-alarm control (Gu et al., 2025; Park et al., 2026; Huang et al., 2026). However, conjunctive fusion can suppress alarms when an action branch remains quiet (Römer et al., 2025), whereas disjunctive fusion changes the combined alarm event whose error must be controlled (Agia et al., 2025). Because instruction–execution mismatch need not produce an action-side anomaly, CALM allows spatial evidence alone to trigger assistance. We accumulate the spatial support deficits and use episode-level split conformal calibration (Angelopoulos & Bates, 2023; Agia et al., 2025) to control unnecessary interruptions over otherwise successful autonomous episodes.

The contributions of this work are summarized in the following aspects.

- We introduce an instruction-conditioned Bayesian spatial readout of a frozen VLA, using execution-endpoint and abstention supervision to construct physically grounded monitoring evidence.
- We develop a sequential intervention rule that combines spatial support deficits with episode-level conformal calibration, providing finite-sample marginal control of unnecessary interruptions under exchangeability.
- Experiments on LIBERO-10 across three VLA backbones ($\pi_{0.5}$, π_0, and OpenVLA (Physical Intelligence et al., 2025; Black et al., 2025; Kim et al., 2024)) show that CALM achieves the best overall performance in instruction–execution mismatch detection among the evaluated methods.

2 Problem Formulation

This work studies runtime detection of instruction-following failures in pretrained VLA policies for robotic systems. We index recorded training episodes by $e = 1, \dots, E$ and use $*$ to denote a new test episode. Each autonomous episode contains at most T replanning steps and ends upon task completion or termination. Let $T_e \leq T$ denote the number of replanning steps in episode e under autonomous execution. At each replan $t = 1, \dots, T_e$, the frozen VLA receives $\mathbf{x}_{e,t} := (\mathbf{v}_{e,t}, \ell_e, \mathbf{s}_{e,t})$, where $\mathbf{v}_{e,t}$, ℓ_e, and $\mathbf{s}_{e,t}$ denote the visual observation, language instruction, and proprioceptive state, respectively. The frozen policy $\pi_{\mathrm{VLA}}(\cdot \mid \mathbf{x}_{e,t})$ generates robot actions for execution. For spatial supervision, we retain the spatial arrangement of the VLA visual patch representations as an $H \times W$ grid (Kim et al., 2024; Zhang et al., 2026a). Let $\mathbf{r}_{e,t}$ denote the recorded end-effector position after a fixed execution horizon following replan (e, t). Using the known camera calibration, $\Pi_{\mathrm{cam}}(\cdot)$ projects $\mathbf{r}_{e,t}$ onto the image plane (Zhang et al., 2026c; Argus et al., 2025), and $\mathcal{G}(\cdot)$ maps the projected point to its corresponding spatial cell. We define $c_{e,t} := \mathcal{G}(\Pi_{\mathrm{cam}}(\mathbf{r}_{e,t})) \in \{1, \dots, HW\}$ for a spatial target and reserve $c_{e,t} = 0$ for abstention. Each episode is further assigned a binary label $b_e \in \{0, 1\}$ which indicates a successful autonomous episode ($b_e = 0$) or an episode requiring assistance ($b_e = 1$). Assistance-required episodes include ordinary execution failures and instruction-related cases involving goal replacement, absent targets, or ambiguous instructions. The resulting sequence of observations and robot executions forms the complete autonomous rollout τ_e.

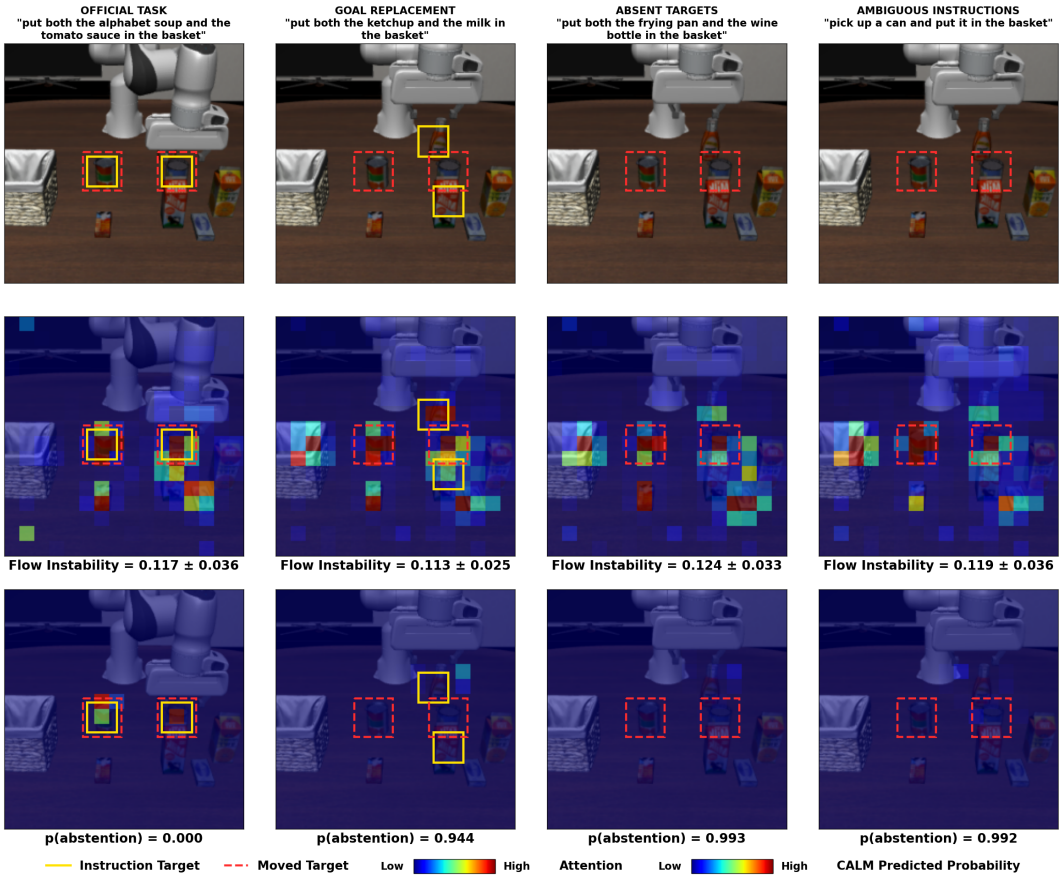


Figure 1: **Instruction–execution mismatch and transferable abstention in CALM.** Columns show the official instruction, goal replacement, absent-target, and ambiguous-instruction conditions in the same familiar scene. Yellow boxes mark instruction targets where specified and visible; red dashed boxes mark objects actually manipulated by the VLA. The top row shows familiar execution persisting under altered instructions. The middle row shows instruction-sensitive attention maps, while *Flow Instability*, the flow-trajectory score of Rao et al. (2026) used here as a proxy for action uncertainty, remains similar across conditions. The official instruction yields concentrated spatial support and negligible abstention, whereas the mismatch conditions show dispersed spatial support without a clear localized prediction.

At runtime, the corresponding test quantities are $\mathbf{x}_{*,t} := (\mathbf{v}_{*,t}, \ell_*, \mathbf{s}_{*,t})$, τ_*, T_*, and b_*. At each replan $(*, t)$, the monitor decides $d_{*,t} \in \{\text{EXECUTE}, \text{ASK}\}$ using information available up to that replan. Here, EXECUTE continues autonomous execution, while ASK requests external assistance and interrupts execution. For an assistance-required episode ($b_* = 1$), the objective is to issue ASK before the episode terminates. For an otherwise successful autonomous episode ($b_* = 0$), the objective is to maintain EXECUTE throughout execution and allow the task to complete without unnecessary intervention. The complete autonomous rollout τ_* and its length T_* refer to execution without intervention; deployment may stop earlier at the first ASK. At replan $(*, t)$, an *instruction–execution mismatch* occurs when the physical continuation favored by the frozen VLA is incompatible with the current instruction ℓ_*. We study instruction–execution mismatch cases under goal-replacement, absent-target, and ambiguous-instruction conditions. Assistance is needed when execution follows the wrong goal, the requested object is missing, or the intended target is unclear.

Related works, challenges and gaps. Familiar and stable execution can conceal a mismatch. Action-based monitors measure uncertainty or temporal consistency (Agia et al., 2025; Römer et al., 2026; Rao et al., 2026), while representation-based monitors use novelty as failure evidence (Römer et al., 2025; Seligmann et al., 2026). These signals may respond weakly when the scene and motion remain familiar but the instruction changes. As shown in Figure 1, the VLA continues to manipulate familiar objects, while Flow Instability, our proxy for action-generation uncertainty (Rao et al., 2026), changes

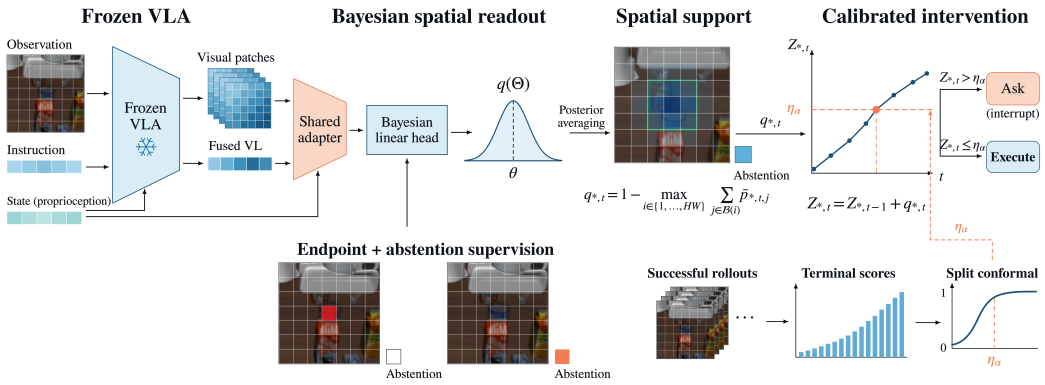


Figure 2: Overview of CALM.

little. The attention maps nevertheless respond to the instruction, suggesting that useful monitoring information remains in the VL representations.

Existing approaches exploit instruction–execution alignment with separate instruction–action verifiers (Kwok et al., 2026), external VLMs judging observed behavior (Grislain et al., 2026), or verification of predicted outcomes (Wu et al., 2025; 2026). These approaches introduce additional inference, while VLM-based verification faces a representational gap in interpreting low-level action consequences (Wu et al., 2025). Outcome-based judgments also depend on the accuracy of world-model or simulation predictions (Wu et al., 2025; 2026). ProbeAct instead reads object positions from internal VLA features, but combines them with robot–object kinematics to detect grasping and placement failures (Zhang et al., 2026a). Its intervention criterion thus remains tied to physical execution abnormalities, which may be absent when the policy smoothly executes the wrong instruction (Figure 1). The challenge is to make instruction-conditioned spatial support itself informative for intervention.

Turning monitoring evidence into intervention also requires calibration. Conformal methods support calibrated assistance and abstention (Ren et al., 2023; Yuan et al., 2026; Singh et al., 2026), as well as trajectory-level false-alarm control (Gu et al., 2025; Park et al., 2026; Huang et al., 2026). The alarm rule nevertheless determines whether available evidence can trigger intervention: conjunctive fusion can suppress alarms when the action branch remains quiet (Römer et al., 2025), whereas component-level guarantees do not automatically hold at the same level after disjunctive fusion (Agia et al., 2025). The challenge is therefore to translate instruction-sensitive evidence into assistance requests without requiring action-side anomalies, while controlling unnecessary interruptions over the entire episode. Appendix B provides a broader discussion of related works.

3 Method

To address these challenges, CALM uses a variational Bayesian readout to predict execution endpoints and abstention from frozen VLA representations with *calibrated uncertainty*. We accumulate spatial support deficits and request assistance at the first threshold crossing. Episode-level split conformal calibration sets this threshold from terminal scores of successful rollouts, providing marginal control of unnecessary interruptions under episode-level exchangeability.

3.1 Bayesian Spatial Prediction

Our goal here is to model the mapping from the inherent visual and VL features to the spatial supervision label $c_{e,t}$. Successful execution endpoints link these instruction-sensitive features to physical outcomes, enabling a spatial readout of execution support under the current instruction. Specifically, at replan (e, t) of a recorded episode, let $\mathbf{f}_i(\mathbf{v}_{e,t})$ denote the frozen visual feature at spatial location $i \in \{1, \ldots, HW\}$, and let $\mathbf{h}_{\mathrm{VL}}(\mathbf{x}_{e,t})$ denote the fused vision–language representation from the same VLA forward pass. A shared adapter $\phi_{\boldsymbol{\omega}}$ with trainable parameters $\boldsymbol{\omega}$ combines the local visual feature, the fused VL representation, and the proprioceptive state to yield a feature

representation for a patch i on the $H \times W$ grid

$$\boldsymbol{\phi}_{e,t,i} := \boldsymbol{\phi}_{\boldsymbol{\omega}} \left(\mathbf{f}_i(\mathbf{v}_{e,t}), \mathbf{h}_{\mathrm{VL}}(\mathbf{x}_{e,t}), \mathbf{s}_{e,t} \right), \quad i = 1, \ldots, HW, \tag{1}$$

In addition, when the current VLA state does not provide sufficient support for a particular physical location (namely, $i = 0$), the feature representation is given by $\boldsymbol{\phi}_{e,t,0} := \boldsymbol{\phi}_{\boldsymbol{\omega}} \left(\mathbf{h}_{\mathrm{VL}}(\mathbf{x}_{e,t}), \mathbf{s}_{e,t} \right)$.

Then, we use a shared linear prediction head across spatial locations, which yields the latent logits(Long et al., 2015)

$$z_{e,t,0} = \boldsymbol{\theta}_0^\top \boldsymbol{\phi}_{e,t,0}, \qquad z_{e,t,i} = \boldsymbol{\theta}^\top \boldsymbol{\phi}_{e,t,i}, \quad i = 1, \ldots, HW, \tag{2}$$

where $\boldsymbol{\theta}$ denote the shared weight vector for the HW spatial outputs and $\boldsymbol{\theta}_0$ is the weight vector for abstention. For principled *uncertainty* reasoning, we place independent Gaussian priors on $\boldsymbol{\Theta} := (\boldsymbol{\theta}, \boldsymbol{\theta}_0)$, namely,

$$p(\boldsymbol{\Theta}) = \mathcal{N}\left(\boldsymbol{\theta}; \mathbf{0}, \sigma_\theta^2 \mathbf{I}\right) \mathcal{N}\left(\boldsymbol{\theta}_0; \mathbf{0}, \sigma_\theta^2 \mathbf{I}\right). \tag{3}$$

where σ_θ is the prior variance.

The logit vector $\mathbf{z}_{e,t} := [z_{e,t,0}, z_{e,t,1}, \ldots, z_{e,t,HW}]^\top$ is further linked with the label $c_{e,t} \in \{0, 1, \ldots, HW\}$ via the categorical likelihood as

$$p\left(c_{e,t} \mid \mathbf{x}_{e,t}, \boldsymbol{\Theta}, \boldsymbol{\omega}\right) = \frac{\exp(z_{e,t,c_{e,t}})}{\sum_{j=0}^{HW} \exp(z_{e,t,j})}. \tag{4}$$

For successful executions under official instructions, $c_{e,t} \in \{1, \ldots, HW\}$ is the grid cell containing the projected end-effector position after the execution horizon. Absent-target examples use $c_{e,t} = 0$ to supervise abstention. Goal-replacement, ambiguous-instruction, and ordinary execution-failure episodes are excluded from training. Let $\mathcal{E}_{\mathrm{tr}} \subseteq \{1, \ldots, E\}$ collect the indices of the training episodes.

3.2 Variational Bayesian Training

Given the training episodes $\{\tau_e\}_{e \in \mathcal{E}_{\mathrm{tr}}}$, the goal is to find the feature mapping $\boldsymbol{\phi}_{\boldsymbol{\omega}}(\cdot)$ along with the posterior of $\boldsymbol{\Theta}$. For scalability, we approximate the last-layer posterior with the factorized variational distribution $q(\boldsymbol{\Theta}) = q(\boldsymbol{\theta}) q(\boldsymbol{\theta}_0)$, where

$$q(\boldsymbol{\theta}) = \mathcal{N}\left(\boldsymbol{\theta}; \boldsymbol{\mu}, \boldsymbol{\Sigma}\right), \qquad q(\boldsymbol{\theta}_0) = \mathcal{N}\left(\boldsymbol{\theta}_0; \boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0\right). \tag{5}$$

We rely on the varitional (V) BLL training that jointly seek all the parameters by minimizing the following negative evidence lower bound (ELBO) (Harrison et al., 2024)

$$\begin{aligned} \mathcal{L}\left(\boldsymbol{\omega}, \boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0\right) = -\mathbb{E}_{q(\boldsymbol{\Theta})} \left[\sum_{e \in \mathcal{E}_{\mathrm{tr}}} \sum_{t=1}^{T_e} \log p\left(c_{e,t} \mid \mathbf{x}_{e,t}, \boldsymbol{\Theta}, \boldsymbol{\omega}\right) \right] \\ + \mathrm{KL}\left(q(\boldsymbol{\theta}) \,\|\, p(\boldsymbol{\theta})\right) + \mathrm{KL}\left(q(\boldsymbol{\theta}_0) \,\|\, p(\boldsymbol{\theta}_0)\right). \end{aligned} \tag{6}$$

where $\mathrm{KL}(\cdot\|\cdot)$ denotes the Kullback-Leibler divergence between two distributions, which has a closed-form expression for two Gaussians.

After training, we fix $\boldsymbol{\omega}$, $q(\boldsymbol{\Theta})$, and all other monitoring choices before calibration. Given test input $\mathbf{x}_{*,t}$ at runtime, we obtain the posterior predictive probability for pointing at grid j (namely $c_{*,t} = j, j = 0, 1, \ldots, HW$) via Monte-Carlo sampling as

$$\begin{aligned} \bar{p}_{*,t,j} := \mathbb{P}(c_{*,t} = j | \mathbf{x}_{*,t}, \{\tau_e\}_{e \in \mathcal{E}_{\mathrm{tr}}}) &= \int p\left(c_{*,t} \mid \mathbf{x}_{*,t}, \boldsymbol{\Theta}, \boldsymbol{\omega}\right) q(\boldsymbol{\Theta}) d\boldsymbol{\Theta} \\ &\approx \frac{1}{S} \sum_{s=1}^{S} \frac{\exp(z_{*,t,j}^{(s)})}{\sum_{j'=0}^{HW} \exp(z_{*,t,j'}^{(s)})}. \end{aligned} \tag{7}$$

where $z_{*,t,j}^{(s)}$ denotes the logit j obtained from the sampled parameters $\boldsymbol{\Theta}^{(s)} \sim q(\boldsymbol{\Theta})$, $s = 1, \ldots, S$. Averaging over the variational posterior carries epistemic uncertainty in the last-layer parameters into these calibrated predictions.

3.3 SEQUENTIAL MONITORING AND CALIBRATION

CALM leverages the calibrated $\bar{p}_{*,t,j}$ (7) for sequential runtime monitoring. Specifically, with $\mathcal{B}(i) \subseteq \{1, \ldots, HW\}$ denoting a fixed local spatial neighborhood centered at location i, we define the monitoring score as the spatial support deficit

$$q_{*,t} = 1 - \max_{i \in \{1,\ldots,HW\}} \sum_{j \in \mathcal{B}(i)} \bar{p}_{*,t,j}. \tag{8}$$

The maximum neighborhood mass measures support for a localized physical continuation. Its complement $q_{*,t} \in [0, 1]$ increases when predictive mass is dispersed across locations or assigned to abstention. Successful-endpoint supervision connects this support to physical execution, while absent-target supervision teaches abstention when the requested target is unavailable. We use the resulting support deficit as a learned proxy for *instruction-execution mismatch*.

Then, we accumulate the monitoring scores using a one-sided cumulative-sum (CUSUM) statistic (Page, 1954). Starting from $Z_{*,0} = 0$, we recursively update

$$Z_{*,t} = Z_{*,t-1} + q_{*,t}. \tag{9}$$

Since $q_{*,t} \geq 0$, the cumulative statistic is nondecreasing throughout each episode.

Calibration by conformal prediction (CP). We will also need a set of calibration episodes $\mathcal{D}_{\text{cal}}^{\text{ep}} := \{\tau_e : e \in \mathcal{E}_{\text{cal}}\}$, where $\mathcal{E}_{\text{cal}}$ collects the indices of the N_{cal} complete successful autonomous rollouts held out from training, with $b_e = 0$. For each calibration episode $e \in \mathcal{E}_{\text{cal}}$, we apply the same monitoring score and CUSUM update based on (7)-(9) to obtain its terminal cumulative score Z_{e,T_e}. Then, we will apply episode-level split conformal calibration to these terminal scores (Agia et al., 2025; Angelopoulos & Bates, 2023). Specifically, let $Z_{(k)}^{\text{cal}}$ denote the k-th smallest value among $\{Z_{e_i,T_{e_i}}\}_{i=1}^{N_{\text{cal}}}$, with $Z_{(N_{\text{cal}}+1)}^{\text{cal}} := +\infty$. For a target unnecessary-interruption level $\alpha \in (0, 1)$, set

$$k_\alpha = \lceil (N_{\text{cal}} + 1)(1 - \alpha) \rceil, \qquad \eta_\alpha = Z_{(k_\alpha)}^{\text{cal}}. \tag{10}$$

At each test replan, the decision is

$$d_{*,t} = \begin{cases} \text{ASK}, & Z_{*,t} > \eta_\alpha, \\ \text{EXECUTE}, & Z_{*,t} \leq \eta_\alpha. \end{cases} \tag{11}$$

Execution is interrupted at the first ASK decision. Before the first ASK, the monitor leaves autonomous execution unchanged. Since the cumulative statistic is nondecreasing, requesting assistance is equivalent to $Z_{*,T_*} > \eta_\alpha$ on the corresponding complete autonomous rollout, where T_* is its length. Algorithm 1 in App. A summarizes the whole process, including readout training, calibration, and online monitoring.

Theoretical guarantees. Below are some theoretical results. Proposition 3.1 bounds the probability of interrupting an otherwise successful autonomous episode, accounting for all replanning decisions within that episode. Corollary 3.2 quantifies the uncertainty in the deployment interruption rate arising from estimating the threshold using a finite set of successful rollouts. The proofs of Proposition 3.1 and Corollary 3.2 are given in Appendix C.1.

Proposition 3.1 (Control of unnecessary interruptions)**.** *Let $\boldsymbol{\omega}$, $q(\boldsymbol{\Theta})$, and all other monitoring choices be fixed before calibration, with Eqs. (8) and (9) used throughout calibration and deployment. Assume that the complete successful calibration rollouts $\tau_{e_1}, \ldots, \tau_{e_{N_{\text{cal}}}}$ and a future successful autonomous test rollout τ_* are exchangeable. For the threshold η_α defined in Eq. (10),*

$$\Pr_{\mathcal{D}_{\text{cal}}^{\text{ep}},\, \tau_*} (\exists\, t \in \{1, \ldots, T_*\} : Z_{*,t} > \eta_\alpha) \leq \alpha. \tag{12}$$

The probability is marginal over the calibration set and the future successful autonomous test rollout.

For a fixed calibration set, let $R(\mathcal{D}_{\text{cal}}^{\text{ep}}) := \Pr(Z_{*,T_*} > \eta_\alpha \mid \mathcal{D}_{\text{cal}}^{\text{ep}})$ denote the interruption rate on a future successful autonomous test episode. Proposition 3.1 guarantees $\mathbb{E}[R] \leq \alpha$. The following corollary characterizes how this rate varies across calibration sets.

Corollary 3.2 (Distribution of the realized interruption rate)*.* Suppose additionally that these calibration and test rollouts are i.i.d., with a continuous distribution of terminal cumulative scores. For $1 \leq k_\alpha \leq N_{\text{cal}}$,

$$R(\mathcal{D}_{\text{cal}}^{\text{ep}}) \sim \text{Beta}\,(N_{\text{cal}} + 1 - k_\alpha,\, k_\alpha)\,, \tag{13}$$

where k_α is defined in Eq. equation 10. Consequently, $\mathbb{E}[R] = (N_{\text{cal}} + 1 - k_\alpha)/(N_{\text{cal}} + 1) \leq \alpha$ and $\text{Var}[R] = \frac{k_\alpha(N_{\text{cal}}+1-k_\alpha)}{(N_{\text{cal}}+1)^2(N_{\text{cal}}+2)}$. If $k_\alpha = N_{\text{cal}} + 1$, then $\eta_\alpha = +\infty$ and $R = 0$ almost surely.

4 EXPERIMENTS

We evaluate **Q1.** Whether instruction-conditioned spatial support enables effective runtime monitoring across policy families; **Q2.** How abstention transfers beyond its training conditions; and **Q3.** How calibration controls the intervention trade-off. Our main experiments evaluate the spatial score in Eq. 8 on the frozen flow-matching policy $\pi_{0.5}$ (Physical Intelligence et al., 2025) and autoregressive OpenVLA (Kim et al., 2024). Additional results on π_0 (Black et al., 2025) are provided in App. G.1. App. D evaluates the action-uncertainty extension for improved detection of execution failures under the original instructions.

4.1 EXPERIMENTAL SETUP

Experimental setup. We evaluate on the ten LIBERO-10 tasks (Liu et al., 2023), using 570 rollouts per policy spanning original instructions, goal replacements, absent targets, and ambiguous instructions. The spatial readout is trained only on successful rollouts under the original instructions and absent-target examples. Goal replacements and ambiguous instructions are reserved for evaluation. Thresholds are calibrated on $N_{\text{cal}} = 50$ held-out successful rollouts at $\alpha = 0.15$. We report mean $\pm$ standard deviation over three seeds. Full experimental settings are provided in App. E.

Baselines. We compare against runtime failure predictors, uncertainty scores, representation-based OOD detectors, and vision–language verifiers. Baselines follow their respective training and calibration procedures; all methods are evaluated on the same test episodes for each policy and seed. Full settings and baseline descriptions are provided in App. E. We also evaluate logistic regression and MLP classifiers as intervention-rule ablations (App. H.2).

Metrics. An episode is flagged if the monitor issues ASK at any replan. FPR is the flagged fraction of successful test episodes. TPR is computed over pooled original-task execution failures, disobeyed goal replacements, ambiguous, and held-out absent-target episodes. We also report balanced accuracy, time-weighted accuracy (TWA), and normalized detection time. Detection time averages the normalized first-alarm time over detected positives. Metric definitions are provided in App. E.1.

4.2 MONITORING INSTRUCTION–EXECUTION MISMATCH

Tables 1 and 2 show that CALM achieves the highest TPR, balanced accuracy, and TWA among methods within the empirical FPR budget on both the flow-matching $\pi_{0.5}$ policy and autoregressive OpenVLA. Compared with SAFE, it detects more assistance-required episodes while interrupting fewer successful executions on both policies. GPLA alarms earlier on detected episodes, but CALM detects a larger fraction of assistance-required episodes and achieves higher TWA. These results show that instruction-conditioned spatial support provides effective runtime monitoring across different action-generation architectures. **Note.** FM-accel (Rao et al., 2026) and DiffDAgger (Lee et al., 2025) rely on flow trajectories and denoising losses, respectively, which are unavailable from OpenVLA's autoregressive action head. This advantage also holds on π_0, where CALM achieves the highest TPR, balanced accuracy, and TWA within the empirical FPR budget (App. G.1, Table 9).

Transfer of abstention. Table 3 shows complete detection of the tested goal-replacement and ambiguous-instruction episodes, although neither condition appears in readout training. Training with successful execution endpoints and absent-target abstention therefore produces an intervention response that generalizes beyond the supervised instruction conditions.

4.3 UNDERSTANDING THE SPATIAL READOUT

Instruction response at matched states. With recorded observations, proprioceptive states, and denoising noise fixed, goal replacement increases spatial support deficits and abstention (App. F.1, Table 6). Most new assistance requests arise on originally successful trajectories, which CALM generally allows to continue under the original instruction but flags after goal replacement. Action uncertainty, as indicated by Flow Instability, changes little, while the action-chunk change induced by instruction replacement is smaller than that between consecutive replans. The spatial readout thus provides instruction-sensitive evidence for intervention even when the action-side response remains weak.

Table 1: Instruction–execution mismatch detection on the $\pi_{0.5}$. Bold and underline indicate the best and second-best results among rows whose mean empirical FPR is at most $\alpha = 0.15$. – denotes an unavailable quantity.

Method	FPR ↓	TPR ↑	Bal. Acc. ↑	TWA ↑	Det. Time ↓
Within FPR budget (≤ 0.15)					
CALM	$0.086_{\pm 0.006}$	$\mathbf{0.970}_{\pm 0.012}$	$\mathbf{0.942}_{\pm 0.003}$	$\mathbf{0.894}_{\pm 0.004}$	$\underline{0.099}_{\pm 0.006}$
SAFE	$0.133_{\pm 0.029}$	$\underline{0.862}_{\pm 0.132}$	$\underline{0.864}_{\pm 0.057}$	$0.787_{\pm 0.051}$	$0.176_{\pm 0.118}$
Hide-and-Seek	$0.121_{\pm 0.057}$	$0.668_{\pm 0.064}$	$0.773_{\pm 0.060}$	$0.676_{\pm 0.036}$	$0.288_{\pm 0.058}$
ActProbe	$0.083_{\pm 0.064}$	$0.328_{\pm 0.218}$	$0.623_{\pm 0.083}$	$0.559_{\pm 0.038}$	$0.359_{\pm 0.069}$
CoVer	$0.113_{\pm 0.059}$	$0.827_{\pm 0.135}$	$0.857_{\pm 0.039}$	$0.732_{\pm 0.029}$	$0.306_{\pm 0.026}$
GPLA	$0.102_{\pm 0.015}$	$0.804_{\pm 0.079}$	$0.851_{\pm 0.039}$	$\underline{0.820}_{\pm 0.043}$	$\mathbf{0.079}_{\pm 0.018}$
VLA-FAIL	$0.071_{\pm 0.061}$	$0.330_{\pm 0.158}$	$0.630_{\pm 0.051}$	$0.544_{\pm 0.019}$	$0.547_{\pm 0.081}$
RND-OE	$0.145_{\pm 0.009}$	$0.198_{\pm 0.028}$	$0.527_{\pm 0.017}$	$0.502_{\pm 0.015}$	$0.248_{\pm 0.014}$
logpZO	$0.121_{\pm 0.006}$	$0.248_{\pm 0.035}$	$0.564_{\pm 0.015}$	$0.524_{\pm 0.012}$	$0.322_{\pm 0.023}$
Rel. Mahalanobis	$0.117_{\pm 0.081}$	$0.612_{\pm 0.084}$	$0.747_{\pm 0.029}$	$0.614_{\pm 0.024}$	$0.440_{\pm 0.042}$
Deep k-NN	$0.125_{\pm 0.031}$	$0.397_{\pm 0.085}$	$0.636_{\pm 0.028}$	$0.539_{\pm 0.007}$	$0.486_{\pm 0.004}$
FOREWARN	$0.063_{\pm 0.048}$	$0.006_{\pm 0.006}$	$0.471_{\pm 0.022}$	$0.469_{\pm 0.024}$	–
SEAL	$0.074_{\pm 0.037}$	$0.127_{\pm 0.003}$	$0.526_{\pm 0.017}$	$0.515_{\pm 0.017}$	$0.177_{\pm 0.008}$
I-FailSense	$0.043_{\pm 0.024}$	$0.063_{\pm 0.009}$	$0.510_{\pm 0.009}$	$0.498_{\pm 0.009}$	$0.386_{\pm 0.039}$
CLIP	$0.086_{\pm 0.018}$	$0.015_{\pm 0.006}$	$0.464_{\pm 0.005}$	$0.457_{\pm 0.009}$	–
Above FPR budget (> 0.15)					
ProbeAct	$0.219_{\pm 0.047}$	$0.189_{\pm 0.080}$	$0.485_{\pm 0.022}$	$0.435_{\pm 0.008}$	$0.529_{\pm 0.036}$
FIPER	$0.332_{\pm 0.066}$	$0.614_{\pm 0.054}$	$0.641_{\pm 0.012}$	$0.523_{\pm 0.023}$	$0.385_{\pm 0.020}$
ACE	$0.187_{\pm 0.071}$	$0.496_{\pm 0.057}$	$0.655_{\pm 0.008}$	$0.537_{\pm 0.018}$	$0.476_{\pm 0.013}$
DiffDAgger	$0.156_{\pm 0.076}$	$0.468_{\pm 0.051}$	$0.656_{\pm 0.017}$	$0.554_{\pm 0.026}$	$0.435_{\pm 0.006}$
TC	$0.203_{\pm 0.032}$	$0.401_{\pm 0.086}$	$0.599_{\pm 0.032}$	$0.494_{\pm 0.016}$	$0.526_{\pm 0.019}$
RND-A	$0.223_{\pm 0.041}$	$0.205_{\pm 0.056}$	$0.491_{\pm 0.010}$	$0.435_{\pm 0.006}$	$0.559_{\pm 0.035}$
Similarity	$0.152_{\pm 0.039}$	$0.101_{\pm 0.030}$	$0.474_{\pm 0.005}$	$0.456_{\pm 0.012}$	$0.367_{\pm 0.048}$
FM-accel	$0.168_{\pm 0.049}$	$0.548_{\pm 0.064}$	$0.690_{\pm 0.010}$	$0.580_{\pm 0.011}$	$0.400_{\pm 0.020}$
UPS	$0.848_{\pm 0.019}$	$0.749_{\pm 0.010}$	$0.451_{\pm 0.006}$	$0.376_{\pm 0.009}$	$0.201_{\pm 0.041}$
KnowNo	$0.891_{\pm 0.028}$	$0.636_{\pm 0.012}$	$0.373_{\pm 0.010}$	$0.200_{\pm 0.012}$	$0.544_{\pm 0.006}$

Spatial-monitor ablations. Table 7 (App. F.1) shows that the Bayesian head improves mean TPR, balanced accuracy, and TWA over the deterministic last layer while reducing mean FPR. Removing the patch grid increases unnecessary interruptions and delays intervention, reducing TWA. Using abstention probability alone lowers TPR and balanced accuracy with similar mean TWA, indicating that spatial concentration provides information beyond abstention. Conversely, removing the abstention output increases FPR and delays intervention, reducing both balanced accuracy and TWA despite a slightly higher TPR. Table 8 (App. F.1) further shows that the readout without the abstention output produces dispersed spatial predictions under mismatch conditions and concentrated predictions on successful episodes. All these results support complementary roles for localized spatial concentration and explicit abstention in identifying episodes that require assistance.

4.4 Calibration and Further Evaluations

Calibration. The spatial monitor retains high detection coverage under stricter interruption budgets, with similar TWA across the more conservative settings (App. H.1, Table 10).

Intervention-rule ablations. For instruction–execution mismatch detection, CALM achieves lower FPR and higher TWA than functional conformal bands. Against direct intervention classifiers, CALM achieves higher TPR and lower FPR than variants trained without failure labels, and higher TWA with far fewer unnecessary interruptions than variants trained with failure labels (App. H.2, Table 11).

Complementary action uncertainty. Table 4 in App. D.2, shows that the extension improves detection of execution failures under the original instructions, achieving a higher execution-failure

Table 2: Instruction–execution mismatch detection on the OpenVLA

Method	FPR ↓	TPR ↑	Bal. Acc. ↑	TWA ↑	Det. Time ↓
Within FPR budget (≤ 0.15)					
CALM	$0.141_{\pm 0.105}$	$\mathbf{0.899}_{\pm 0.023}$	$\mathbf{0.879}_{\pm 0.041}$	$\mathbf{0.767}_{\pm 0.033}$	$0.250_{\pm 0.032}$
SAFE	$0.149_{\pm 0.118}$	$\underline{0.807}_{\pm 0.011}$	$\underline{0.829}_{\pm 0.059}$	$\underline{0.709}_{\pm 0.085}$	$0.298_{\pm 0.067}$
ActProbe	$0.060_{\pm 0.054}$	$0.466_{\pm 0.079}$	$0.703_{\pm 0.025}$	$0.603_{\pm 0.021}$	$0.430_{\pm 0.024}$
GPLA	$0.127_{\pm 0.096}$	$0.523_{\pm 0.084}$	$0.698_{\pm 0.012}$	$0.684_{\pm 0.016}$	$\mathbf{0.055}_{\pm 0.027}$
VLA-FAIL	$0.112_{\pm 0.048}$	$0.380_{\pm 0.148}$	$0.634_{\pm 0.053}$	$0.552_{\pm 0.027}$	$0.442_{\pm 0.037}$
ACE	$0.118_{\pm 0.086}$	$0.341_{\pm 0.067}$	$0.612_{\pm 0.047}$	$0.517_{\pm 0.039}$	$0.557_{\pm 0.024}$
Similarity	$0.121_{\pm 0.063}$	$0.288_{\pm 0.081}$	$0.584_{\pm 0.013}$	$0.516_{\pm 0.009}$	$0.475_{\pm 0.034}$
logpZO	$0.140_{\pm 0.080}$	$0.655_{\pm 0.032}$	$0.757_{\pm 0.024}$	$0.632_{\pm 0.024}$	$0.383_{\pm 0.022}$
Rel. Mahalanobis	$0.118_{\pm 0.018}$	$0.760_{\pm 0.020}$	$0.821_{\pm 0.006}$	$0.637_{\pm 0.007}$	$0.484_{\pm 0.013}$
Deep k-NN	$0.144_{\pm 0.129}$	$0.729_{\pm 0.029}$	$0.792_{\pm 0.050}$	$0.630_{\pm 0.052}$	$0.446_{\pm 0.013}$
FOREWARN	$0.094_{\pm 0.123}$	$0.045_{\pm 0.074}$	$0.476_{\pm 0.026}$	$0.477_{\pm 0.036}$	$0.432_{\pm 0.196}$
SEAL (7B)	$0.091_{\pm 0.101}$	$0.128_{\pm 0.005}$	$0.518_{\pm 0.052}$	$0.506_{\pm 0.050}$	$0.206_{\pm 0.026}$
I-FailSense	$0.033_{\pm 0.058}$	$0.085_{\pm 0.016}$	$0.526_{\pm 0.021}$	$0.521_{\pm 0.024}$	$\underline{0.127}_{\pm 0.052}$
CLIP	$0.066_{\pm 0.057}$	$0.033_{\pm 0.020}$	$0.483_{\pm 0.019}$	$0.467_{\pm 0.029}$	–
Above FPR budget (> 0.15)					
Hide-and-Seek	$0.178_{\pm 0.062}$	$0.879_{\pm 0.028}$	$0.851_{\pm 0.022}$	$0.814_{\pm 0.029}$	$0.083_{\pm 0.016}$
CoVer	$0.196_{\pm 0.111}$	$0.732_{\pm 0.065}$	$0.768_{\pm 0.050}$	$0.652_{\pm 0.056}$	$0.321_{\pm 0.057}$
ProbeAct	$0.278_{\pm 0.068}$	$0.064_{\pm 0.044}$	$0.393_{\pm 0.027}$	$0.372_{\pm 0.030}$	$0.643_{\pm 0.024}$
FIPER	$0.250_{\pm 0.180}$	$0.711_{\pm 0.048}$	$0.730_{\pm 0.067}$	$0.578_{\pm 0.067}$	$0.429_{\pm 0.030}$
RND-OE	$0.171_{\pm 0.171}$	$0.603_{\pm 0.083}$	$0.716_{\pm 0.058}$	$0.594_{\pm 0.061}$	$0.409_{\pm 0.026}$
RND-A	$0.172_{\pm 0.066}$	$0.296_{\pm 0.013}$	$0.562_{\pm 0.026}$	$0.477_{\pm 0.031}$	$0.576_{\pm 0.019}$
UPS	$1.000_{\pm 0.000}$	$1.000_{\pm 0.000}$	$0.500_{\pm 0.000}$	$0.500_{\pm 0.000}$	$0.000_{\pm 0.000}$
KnowNo	$1.000_{\pm 0.000}$	$1.000_{\pm 0.000}$	$0.500_{\pm 0.000}$	$0.414_{\pm 0.002}$	$0.173_{\pm 0.004}$

Table 3: Instruction conditions for the spatial monitor on $\pi_{0.5}$. Endpoint and abstention denote supervision on separate training examples;

Test condition	Training supervision	Metric	Rate
Official success	Endpoint	FPR	0.086 ± 0.006
Goal replacement	None	TPR	1.000 ± 0.000
Absent target	Abstention	TPR	1.000 ± 0.000
Ambiguous instruction	None	TPR	1.000 ± 0.000

TPR than spatial monitoring alone. It also improves pooled TPR and balanced accuracy while maintaining a low FPR.

Deployment cost. Compared with methods requiring additional VL inference, CALM achieves stronger detection at lower computational cost (Tables 1 and 12). It also improves detection over computationally lighter action-based monitors while keeping its overhead small relative to policy inference.

5 Conclusions

This paper introduced CALM, a runtime monitor for instruction–execution mismatch in pretrained VLA policies. Specifically, CALM relies on instruction-conditioned spatial support inferred from frozen VLA representations and then leverages conformal calibration to decide on externel assistance. Across $\pi_{0.5}$, π_0, and OpenVLA, this spatial monitoring signal achieves the highest TPR, balanced accuracy, and TWA among the evaluated methods within the empirical FPR budget, demonstrating its effectiveness across flow-matching and autoregressive policies. Beyond empirical detection performance, episode-level conformal calibration provides finite-sample marginal control of unnecessary interruptions under exchangeability.

Reproducibility Statement

We provide the algorithm in Appendix A, experimental settings and data splits in Appendix E, metric definitions in Appendix E.1, additional ablations in Appendix F, and theoretical proofs in Appendix C to support reproducibility of our results.

Ethics Statement

This work studies runtime monitoring for robotic manipulation in simulation and does not involve human subjects or sensitive personal data.

AI Use Statement

Generative AI tools were used to assist with language editing and manuscript organization.

References

Christopher Agia, Rohan Sinha, Jingyun Yang, Ziang Cao, Rika Antonova, Marco Pavone, and Jeannette Bohg. Unpacking failure modes of generative policies: Runtime monitoring of consistency and progress. In *Proceedings of The 8th Conference on Robot Learning*, volume 270 of *Proceedings of Machine Learning Research*, pp. 689–723, 2025.

Anastasios N. Angelopoulos and Stephen Bates. Conformal prediction: A gentle introduction. *Found. Trends Mach. Learn.*, 16(4):494–591, 2023.

Anastasios N. Angelopoulos, Stephen Bates, Adam Fisch, Lihua Lei, and Tal Schuster. Conformal risk control. In *Proc. Intl. Conf. Learn. Repr.*, 2024.

Max Argus, Jelena Bratulić, Houman Masnavi, Maxim Velikanov, Nick Heppert, Abhinav Valada, and Thomas Brox. cVLA: Towards efficient camera-space VLAs. *arXiv preprint arXiv:2507.02190*, 2025.

Stephen Bates, Anastasios N. Angelopoulos, Lihua Lei, Jitendra Malik, and Michael I. Jordan. Distribution-free, risk-controlling prediction sets. *Journal of the ACM*, 68(6):43:1–43:34, 2021. doi: 10.1145/3478535.

Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Robert Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Lucy Xiaoyang Shi, Laura Smith, James Tanner, Quan Vuong, Anna Walling, Haohuan Wang, and Ury Zhilinsky. π_0: A Vision-Language-Action Flow Model for General Robot Control. In *Robotics: Science and Systems*, LosAngeles, CA, USA, June 2025. doi: 10.15607/RSS.2025.XXI.010.

Xiwen Chen, Zelin Li, Zhiruo Zhou, Huiming Chen, Chenwei Wang, and Xiaojun Zhu. Pointing-VLA: Typed spatial grounding interfaces for vision-language-action manipulation. *arXiv preprint arXiv:2608.23138*, 2026.

Cheng Chi, Siyuan Feng, Yilun Du, Zhenjia Xu, Eric Cousineau, Benjamin Burchfiel, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. In *Robotics: Science and Systems*, 2023.

Nastaran Darabi and Amit Ranjan Trivedi. ProGAL-VLA: Grounded alignment through prospective reasoning in vision-language-action models. In *Proc. IEEE/CVF Conf. Comput. Vision Pattern Recognit. Findings*, pp. 9013–9022, 2026.

Yu Fang, Yuchun Feng, Dong Jing, Jiaqi Liu, Yue Yang, Zhenyu Wei, Daniel Szafir, and Mingyu Ding. When vision overrides language: Evaluating and mitigating counterfactual failures in VLAs. *arXiv preprint arXiv:2602.17659*, 2026.

Senyu Fei, Siyin Wang, Junhao Shi, Zihao Dai, Jikun Cai, Pengfang Qian, Li Ji, Xinzhe He, Shiduo Zhang, Zhaoye Fei, Jinlan Fu, Jingjing Gong, and Xipeng Qiu. LIBERO-Plus: A progressive robustness benchmark for visual-language-action models. In *Proc. of Conf. Comput. Vision Pattern Recognit.*, pp. 38574–38583, 2026.

Clemence Grislain, Hamed Rahimi, Olivier Sigaud, and Mohamed Chetouani. I-FailSense: Towards general robotic failure detection with vision-language models. In *Proc. IEEE Intl. Conf. Robot. Autom.*, 2026.

Qiao Gu, Yuanliang Ju, Shengxiang Sun, Igor Gilitschenski, Haruki Nishimura, Masha Itkina, and Florian Shkurti. SAFE: Multitask failure detection for vision-language-action models. In *Proc. of Adv. Neural Inf. Process. Syst.*, volume 38, 2025.

James Harrison, John Willes, and Jasper Snoek. Variational bayesian last layers. In *Proc. Intl. Conf. Learn. Repr.*, 2024.

Zhanpeng He, Yifeng Cao, and Matei Ciocarlie. Uncertainty comes for free: Human-in-the-loop policies with diffusion models. In *Proc. IEEE Intl. Conf. Robot. Autom.*, 2026.

Bingjia Huang, Xiangyu Li, Xiang Wang, Liang Mi, Zixu Hao, Weijun Wang, Hao Wu, Kun Li, Yunxin Liu, and Ting Cao. ActProbe: Action-space probe for early failure detection of generative robot policies. *arXiv preprint arXiv:2606.08508*, 2026.

Kun Jin, James Harrison, Jiawei Li, Sihan Liu, Jiayi Liu, Randolph Linderman, Yuening Li, Arnab Bhadury, Sourabh Prakash Bansod, Liang Liu, and Jasper Snoek. Efficient analytic uncertainty quantification for multi-modal regression. *arXiv preprint arXiv:2606.25188*, 2026.

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.

Agustinus Kristiadi, Matthias Hein, and Philipp Hennig. Being bayesian, even just a bit, fixes overconfidence in ReLU networks. In *Proc. Intl. Conf. Mach. Learn.*, 2020.

Jacky Kwok, Xilun Zhang, Mengdi Xu, Yuejiang Liu, Azalia Mirhoseini, Chelsea Finn, and Marco Pavone. Scaling verification can be more effective than scaling policy learning for vision-language-action alignment. In *European Conference on Computer Vision (ECCV)*, 2026. To appear.

Sung-Wook Lee, Xuhui Kang, and Yen-Ling Kuo. Diff-DAgger: Uncertainty estimation with diffusion policy for robotic manipulation. In *International Conference on Robotics and Automation (ICRA)*, 2025.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *Proc. Intl. Conf. Learn. Repr.*, 2023.

Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. LIBERO: Benchmarking knowledge transfer for lifelong robot learning. In *Proc. of Adv. Neural Inf. Process. Syst.*, volume 36, pp. 44776–44791, 2023.

Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proc. of Conf. Comput. Vision Pattern Recognit.*, pp. 3431–3440, 2015.

Hong Lu, Hengxu Li, Prithviraj Singh Shahani, Stephanie Herbers, and Matthias Scheutz. Probing a vision-language-action model for symbolic states and integration into a cognitive architecture. *arXiv preprint arXiv:2502.04558*, 2025.

Douglas Morrison, Peter Corke, and Jürgen Leitner. Closing the loop for robotic grasping: A real-time, generative grasp synthesis approach. In *Robotics: Science and Systems*, 2018.

E. S. Page. Continuous inspection schemes. *Biometrika*, 41(1–2):100–115, 1954.

Seongheon Park, Wendi Li, Changdae Oh, Samuel Yeh, Zsolt Kira, Michael Hagenow, and Sharon Li. Hide-and-seek in trajectories: Discovering failure signals for VLA runtime monitoring. *arXiv preprint arXiv:2605.30834*, 2026.

Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, et al. $\pi_{0.5}$: A vision-language-action model with open-world generalization. *arXiv preprint arXiv:2504.16054*, 2025.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proc. Intl. Conf. Mach. Learn.*, 2021.

Ziyang Rao, Yiren Zhao, Weiyu Guo, Ben Fei, Yandong Guo, and Hui Xiong. The geometry of flow-matching uncertainty: A cost-free uncertainty proxy and its application in flow-based VLA failure detection. *arXiv preprint arXiv:2607.27933*, 2026.

Allen Z. Ren, Anushri Dixit, Alexandra Bodrova, Sumeet Singh, Stephen Tu, Noah Brown, Peng Xu, Leila Takayama, Fei Xia, Jake Varley, Zhenjia Xu, Dorsa Sadigh, Andy Zeng, and Anirudha Majumdar. Robots that ask for help: Uncertainty alignment for large language model planners. In *Proceedings of The 7th Conference on Robot Learning*, volume 229 of *Proceedings of Machine Learning Research*, pp. 661–682, 2023.

Jie Ren, Stanislav Fort, Jeremiah Liu, Abhijit Guha Roy, Shreyas Padhy, and Balaji Lakshminarayanan. A simple fix to mahalanobis distance for improving near-OOD detection. *arXiv preprint arXiv:2106.09022*, 2021.

Ralf Römer, Adrian Kobras, Luca Worbis, and Angela P. Schoellig. Failure prediction at runtime for generative robot policies. In *Proc. of Adv. Neural Inf. Process. Syst.*, volume 38, 2025.

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*, 2026.

Florian Seligmann, Emiliyan Gospodinov, Enes Ulas Dincer, and Gerhard Neumann. VLA-FAIL: Efficient task failure detection for finetuned vision-language-action models. *arXiv preprint arXiv:2606.21386*, 2026.

Mohit Shridhar, Lucas Manuelli, and Dieter Fox. CLIPort: What and where pathways for robotic manipulation. In *Proc. Conf. Robot Learn.*, 2022.

Anya Singh, Cabrel Happi, Jai Relan, Varun Nair, and Vidyut Baradwaj. BOKBO (best of k bad options): Calibrated abstention for VLA policies. *arXiv preprint arXiv:2605.30660*, 2026.

Yiyou Sun, Yifei Ming, Xiaojin Zhu, and Yixuan Li. Out-of-distribution detection with deep nearest neighbors. In *International Conference on Machine Learning (ICML)*, 2022.

Zhen Sun, Yongjian Guo, Haoran Sun, Luqiao Wang, Wei Lu, Jiachi Ji, Shengzhe Ji, Junwu Xiong, and Zhijun Meng. Pre-VLA: Preemptive runtime verification for reliable vision-language-action and world-model rollouts. *arXiv preprint arXiv:2605.22446*, 2026.

Yilin Wu, Thomas Tian, Gokul Swamy, and Andrea Bajcsy. From foresight to forethought: VLM-in-the-loop policy steering via latent alignment. In *Robotics: Science and Systems (RSS)*, 2025.

Yilin Wu, Anqi Li, Tucker Hermans, Fabio Ramos, Andrea Bajcsy, and Claudia Pérez-D'Arpino. Do what you say: Steering vision-language-action models via runtime reasoning-action alignment verification. In *Proc. IEEE Intl. Conf. Robot. Autom.*, 2026.

Theodor Wulff, Federico Tavella, Rahul Singh Maharjan, Manith Adikari, and Angelo Cangelosi. Grounding hierarchical vision-language-action models through explicit language-action alignment. In *Proc. IEEE/CVF Conf. Comput. Vision Pattern Recognit. Findings*, pp. 9269–9281, 2026.

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.

Jessie Yuan, Yilin Wu, and Andrea Bajcsy. When to act, ask, or learn: Uncertainty-aware policy steering. In *Robotics: Science and Systems*, 2026.

Andy Zeng, Pete Florence, Jonathan Tompson, Stefan Welker, Jonathan Chien, Maria Attarian, Travis Armstrong, Ivan Krasin, Dan Duong, Vikas Sindhwani, and Johnny Lee. Transporter networks: Rearranging the visual world for robotic manipulation. In *Proc. Conf. Robot Learn.*, 2021.

Fan Zhang, Seongbin Park, Baharan Mirzasoleiman, Shariar Talebi, and Nader Sehatbakhsh. Probe-Act: Probe-guided training-free failure recovery in vision-language-action models. *arXiv preprint arXiv:2606.09740*, 2026a.

Jiachen Zhang, Junnan Nie, Junyi Lao, Wei Cheng, Chenghao Liu, Jiaxin Jiang, and Songfang Huang. What frozen VLAs already know about success: A probing study of value-like structure in foundation robot policies. *arXiv preprint arXiv:2605.28527*, 2026b.

Tianyi Zhang, Haonan Duan, Haoran Hao, Yu Qiao, Jifeng Dai, and Zhi Hou. Grounding actions in camera space: Observation-centric vision-language-action policy. In *Proc. AAAI Conf. Artif. Intell.*, volume 40, pp. 18782–18790, 2026c. doi: 10.1609/aaai.v40i22.38947.

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.

Supplementary File

A ALGORITHM

Algorithm 1 summarizes readout training, episode-level calibration, and online monitoring in CALM.

Algorithm 1 CALM: Training, Calibration, and Online Monitoring

Require: Frozen policy π_{VLA}; labeled training episodes $\mathcal{E}_{\mathrm{tr}}$; held-out successful rollouts $\mathcal{D}_{\mathrm{cal}}^{\mathrm{ep}}$; interruption level α; posterior sample count S; spatial neighborhoods $\{\mathcal{B}(i)\}$.
Ensure: Trained readout, threshold η_α, and online decisions $d_{*,t}$.

S1. Readout training
Learn $\boldsymbol{\omega}$ and $q(\boldsymbol{\Theta})$ by minimizing Eq. 6 on $\mathcal{E}_{\mathrm{tr}}$.
Fix the readout and all scoring choices before calibration.
S2. Episode-level calibration
for each rollout $\tau_e \in \mathcal{D}_{\mathrm{cal}}^{\mathrm{ep}}$ **do**
  Initialize $Z_{e,0} \leftarrow 0$.
  for $t = 1, \ldots, T_e$ **do**
    Compute $\bar{p}_{e,t,j}$ $(j = 0, \ldots, HW)$ using Eq. (7).
    Compute $q_{e,t}$ using Eq. (8) and update $Z_{e,t}$ using Eq. (9).
  end for
end for
Compute η_α from the terminal calibration scores using Eq. (10).
S3. Online monitoring
Initialize $Z_{*,0} \leftarrow 0$ and $t \leftarrow 0$.
while the test episode has not terminated **do**
  $t \leftarrow t + 1$; observe $\mathbf{x}_{*,t}$.
  Run the frozen VLA to obtain its action output and readout representations.
  Compute $\bar{p}_{*,t,j}$ $(j = 0, \ldots, HW)$ using Eq. (7).
  Compute $q_{*,t}$ using Eq. (8) and update $Z_{*,t}$ using Eq. (9).
  Determine $d_{*,t}$ using Eq. (11).
  if $d_{*,t} = \textsc{ask}$ **then**
    Request assistance and interrupt execution.
    break
  else
    Execute the policy output until the next replan or episode termination.
  end if
end while

B RELATED WORKS

This paper is related to the following lines of works.

B.1 RUNTIME FAILURE MONITORING FOR ROBOT POLICIES

Runtime monitoring for generative robot policies draws on signals from action generation and policy representations. Action-space methods measure sample dispersion, temporal consistency, action magnitude, or uncertainty exposed by diffusion and flow trajectories (He et al., 2026; Agia et al., 2025; Huang et al., 2026; Römer et al., 2026; Rao et al., 2026). Representation-based methods use unfamiliar observations or hidden states as failure evidence. FIPER combines representation novelty with action uncertainty, while VLA-FAIL combines representation novelty with action consistency (Römer et al., 2025; Seligmann et al., 2026). Learned monitors such as SAFE and Hide-and-Seek predict failure from frozen policy representations and their temporal evolution (Gu et al., 2025; Park et al., 2026).

Familiar scenes and stable actions can nevertheless conceal instruction–execution mismatch: a policy may smoothly execute a learned routine that is incompatible with its current instruction. Action uncertainty and representation novelty may then remain weak, even though the instruction has changed. Figure 1 illustrates this difficulty: familiar behavior persists and Flow Instability changes little, while attention maps still respond to the instruction. This suggests that useful monitoring information remains in the VL representations.

B.2 Instruction Alignment and Spatial Representations in VLAs

VLA policies can exhibit weak sensitivity to language when familiar visual contexts strongly cue previously learned behaviors. Counterfactual evaluations show continued execution of scene-specific visuomotor routines under altered instructions, together with broader degradation under language and visual perturbations (Fang et al., 2026; Zhou et al., 2025; Fei et al., 2026). One line of work improves language–action coupling during policy training through explicit grounding, prospective reasoning, and alignment objectives (Darabi & Trivedi, 2026; Wulff et al., 2026).

Runtime verification evaluates whether behavior or candidate actions agree with the instruction. CoVer scores instruction–action compatibility among generated proposals (Kwok et al., 2026), while I-FailSense uses an external VLM to judge observed behavior (Grislain et al., 2026). FOREWARN uses a latent world model to predict action consequences for VLM evaluation (Wu et al., 2025), and Do What You Say compares predicted action outcomes with an explicit reasoning plan (Wu et al., 2026). UPS separates semantic task uncertainty from low-level policy feasibility when choosing whether to act, clarify, or intervene (Yuan et al., 2026). Pre-VLA similarly verifies candidate action chunks before execution (Sun et al., 2026).

Separate instruction–action verifiers and external VLM judgments introduce additional inference (Kwok et al., 2026; Grislain et al., 2026). VLM-based verification also faces a representational gap between low-level robot actions and the visual and linguistic inputs used to interpret their consequences (Wu et al., 2025). Outcome-based approaches bridge this gap through predicted consequences, adding world-model inference or simulation and making verification dependent on prediction accuracy (Wu et al., 2025; 2026).

A related body of work shows that frozen VLA representations contain rich spatial and task-relevant structure (Lu et al., 2025; Zhang et al., 2026b; Gu et al., 2025). Intermediate features encode symbolic task states, object properties, spatial relations, and information predictive of future task success. Pointing-VLA exposes points, grounding heatmaps, and visual trajectories as spatial interfaces for manipulation (Chen et al., 2026). Dense spatial prediction has also been used for grasping and language-conditioned manipulation (Morrison et al., 2018; Zeng et al., 2021; Shridhar et al., 2022). ProbeAct decodes task-relevant object positions from VLA features and combines them with robot–object kinematics to detect grasping and placement failures (Zhang et al., 2026a). Although this internal readout avoids an external verifier, its intervention criterion remains tied to physical execution abnormalities, which may be absent when a policy smoothly executes the wrong instruction.

The challenge is to make instruction-conditioned spatial support itself informative for intervention. CALM predicts a distribution over execution endpoint cells and an abstention output through a Bayesian readout of the frozen VLA's representations. Bayesian last-layer inference propagates parameter uncertainty into this spatial prediction by averaging over the last-layer posterior (Kristiadi et al., 2020; Harrison et al., 2024; Jin et al., 2026). The resulting spatial support deficit provides intervention evidence when predictions are dispersed or favor abstention.

B.3 Calibrated Assistance and Intervention

Conformal prediction has been used to calibrate assistance decisions in language-guided robotics. KnowNo constructs calibrated sets of high-level options to determine when a planner should request human clarification (Ren et al., 2023). UPS applies calibrated uncertainty to semantic clarification and low-level intervention (Yuan et al., 2026), while BOKBO uses conformal risk control to abstain when the available VLA action candidates lack sufficient support (Singh et al., 2026).

Conformal methods also support trajectory-level false-alarm control in runtime failure monitoring. SAFE and Hide-and-Seek construct functional conformal bands from successful trajectories (Gu et al., 2025; Park et al., 2026). FIPER calibrates observation- and action-side failure indicators (Römer et al., 2025), while ActProbe calibrates rollout-level maximum failure scores (Huang et al., 2026). Flow-based monitoring combines sequential uncertainty accumulation with conformal calibration (Rao et al., 2026). Distribution-free risk-control methods extend calibration to more general loss functions (Bates et al., 2021; Angelopoulos et al., 2024).

The deployed alarm rule determines whether calibrated evidence can trigger intervention. Conjunctive (AND) fusion can suppress an alarm when the action branch remains quiet (Römer et al., 2025). Disjunctive (OR) fusion allows either branch to trigger, but guarantees for individual components

do not automatically retain the same error level for the combined alarm event (Agia et al., 2025). The challenge is therefore to translate instruction-sensitive evidence into assistance requests without requiring action-side anomalies, while controlling unnecessary interruptions over the entire episode.

CALM accumulates nonnegative spatial support deficits and calibrates their terminal scores on complete successful autonomous rollouts. The resulting rule allows spatial evidence alone to trigger assistance and provides marginal control of unnecessary interruptions over the entire episode under episode-level exchangeability. Under additional i.i.d. and continuity assumptions, we characterize how the realized interruption rate varies across calibration sets.

C Proofs and Additional Analysis

C.1 Episode-Level Conformal Guarantee

Throughout this subsection, the trained adapter parameters $\boldsymbol{\omega}$, the variational posterior $q(\boldsymbol{\Theta})$, and all other monitoring choices are fixed before calibration. The monitoring score and CUSUM statistic are computed using Eqs. (8) and (9) throughout calibration and deployment.

Proof of Proposition 3.1. Let T_* denote the number of replanning steps in the complete autonomous test rollout τ_*. Since $q_{*,t} \geq 0$, the cumulative statistic $Z_{*,t}$ is nondecreasing. The monitor uses information available at the current replan and leaves autonomous execution unchanged before the first ASK. Execution up to the first threshold crossing therefore agrees with the corresponding autonomous rollout. By Eq. (11), the following events are equivalent

$$\{\text{the test requests assistance}\} = \{\exists\, t \in \{1, \dots, T_*\} : Z_{*,t} > \eta_\alpha\} = \{Z_{*,T_*} > \eta_\alpha\}. \tag{14}$$

Under the standard assumption of split CP, the complete successful calibration rollouts $\tau_{e_1}, \dots, \tau_{e_{N_{\text{cal}}}}$ and the future successful autonomous test rollout τ_* are exchangeable. Applying the same fixed monitoring procedure yields exchangeable terminal scores

$$Z_{e_1,T_{e_1}}, \dots, Z_{e_{N_{\text{cal}}},T_{e_{N_{\text{cal}}}}}, Z_{*,T_*}.$$

Each complete rollout, including its length, is treated as one sample, so the episode lengths may differ.

Recall that $k_\alpha = \lceil (N_{\text{cal}} + 1)(1 - \alpha) \rceil$. If $k_\alpha = N_{\text{cal}} + 1$, then $\eta_\alpha = +\infty$ by Eq. (10), and the interruption probability is zero.

Now suppose $1 \leq k_\alpha \leq N_{\text{cal}}$, so that $\eta_\alpha = Z^{\text{cal}}_{(k_\alpha)}$ is the k_α-th smallest calibration terminal score. For the rank argument, break ties using independent uniform random variables attached to the $N_{\text{cal}} + 1$ scores. By exchangeability, the resulting rank of Z_{*,T_*} is uniform on $\{1, \dots, N_{\text{cal}} + 1\}$. Whenever $Z_{*,T_*} > \eta_\alpha$, at least k_α calibration scores are strictly smaller than the test score. Its rank must therefore exceed k_α, giving

$$\Pr_{\mathcal{D}^{\text{ep}}_{\text{cal}},\, \tau_*} (Z_{*,T_*} > \eta_\alpha) \leq \frac{N_{\text{cal}} + 1 - k_\alpha}{N_{\text{cal}} + 1} \leq \alpha. \tag{15}$$

The last inequality follows from the definition of k_α. Combining Eqs. (14) and (15) proves the proposition. □

Realized interruption rate. For $R(\mathcal{D}^{\text{ep}}_{\text{cal}})$ defined in Section 3.3, the law of total expectation gives

$$\mathbb{E}_{\mathcal{D}^{\text{ep}}_{\text{cal}}} [R(\mathcal{D}^{\text{ep}}_{\text{cal}})] = \Pr_{\mathcal{D}^{\text{ep}}_{\text{cal}},\, \tau_*} (Z_{*,T_*} > \eta_\alpha) \leq \alpha. \tag{16}$$

The expectation is over the calibration set, with the trained parameters and monitoring procedure held fixed.

Proof of Corollary 3.2. First suppose $1 \leq k_\alpha \leq N_{\text{cal}}$. Let F denote the continuous CDF of the terminal cumulative score Z_{e,T_e} for a successful autonomous episode. Under the i.i.d. assumption, the probability integral transform gives

$$F(Z_{e_i,T_{e_i}}) \overset{\text{i.i.d.}}{\sim} \text{Uniform}(0,1), \qquad i = 1, \dots, N_{\text{cal}}. \tag{17}$$

Since $\eta_\alpha = Z^{\mathrm{cal}}_{(k_\alpha)}$, $F(\eta_\alpha)$ is the k_α-th order statistic of these uniform variables. Hence,

$$F(\eta_\alpha) \sim \mathrm{Beta}\left(k_\alpha,\, N_{\mathrm{cal}} + 1 - k_\alpha\right). \tag{18}$$

The test episode is independent of the calibration set and its terminal score has CDF F. Therefore,

$$R(\mathcal{D}^{\mathrm{ep}}_{\mathrm{cal}}) = \Pr\left(Z_{*,T_*} > \eta_\alpha \,|\, \mathcal{D}^{\mathrm{ep}}_{\mathrm{cal}}\right) = 1 - F(\eta_\alpha). \tag{19}$$

If $U \sim \mathrm{Beta}(a, b)$, then $1 - U \sim \mathrm{Beta}(b, a)$. Applying this identity to Eq. (18) yields

$$R(\mathcal{D}^{\mathrm{ep}}_{\mathrm{cal}}) \sim \mathrm{Beta}\left(N_{\mathrm{cal}} + 1 - k_\alpha,\, k_\alpha\right), \tag{20}$$

which proves Eq. 13. The mean and variance of this distribution give the expressions stated in Corollary 3.2.

If $k_\alpha = N_{\mathrm{cal}} + 1$, then $\eta_\alpha = +\infty$, so $R(\mathcal{D}^{\mathrm{ep}}_{\mathrm{cal}}) = 0$ almost surely. □

D Extension to Broader Execution Failures

D.1 Incorporating Action Uncertainty

Spatial predictions can remain concentrated even when execution fails under the original task instruction. For flow-matching policies, we augment the spatial monitor with the denoising-acceleration score $\chi_{*,t} \geq 0$ of Rao et al. (2026), computed from the policy's flow, gettting CALM_{ext}. We report this score as Flow Instability in the experiments. The extended monitoring score is

$$q^{\mathrm{ext}}_{*,t} = q_{*,t} + \chi_{*,t}, \tag{21}$$

where $q_{*,t}$ is defined in Eq. (8).

Starting from $Z^{\mathrm{ext}}_{*,0} = 0$, we recursively update

$$Z^{\mathrm{ext}}_{*,t} = Z^{\mathrm{ext}}_{*,t-1} + q^{\mathrm{ext}}_{*,t}. \tag{22}$$

The combined score is nonnegative, so the cumulative statistic is nondecreasing.

We apply the same computation to each complete successful calibration rollout and recalibrate the threshold using its terminal value Z^{ext}_{e,T_e}. Let $Z^{\mathrm{cal,ext}}_{(k)}$ denote the k-th smallest value among $\{Z^{\mathrm{ext}}_{e_i,T_{e_i}}\}_{i=1}^{N_{\mathrm{cal}}}$. With k_α defined in Eq. (10), set $\eta^{\mathrm{ext}}_\alpha = Z^{\mathrm{cal,ext}}_{(k_\alpha)}$, using $Z^{\mathrm{cal,ext}}_{(N_{\mathrm{cal}}+1)} := +\infty$. The extension requests assistance at the first replan satisfying $Z^{\mathrm{ext}}_{*,t} > \eta^{\mathrm{ext}}_\alpha$.

The trained spatial predictor and scoring procedure are fixed before calibration. Under the episode-level exchangeability assumption of Proposition 3.1, the combined terminal scores are exchangeable, and the rank argument in Appendix C.1 gives

$$\Pr_{\mathcal{D}^{\mathrm{ep}}_{\mathrm{cal}},\, \tau_*}\left(\exists\, t \in \{1, \ldots, T_*\} : Z^{\mathrm{ext}}_{*,t} > \eta^{\mathrm{ext}}_\alpha\right) \leq \alpha \tag{23}$$

for a future successful autonomous test rollout. Calibration therefore controls unnecessary interruptions when both spatial and action evidence contribute to the decision.

D.2 Detection of Execution Failures

Table 4 shows that the extended monitor maintains strong overall performance when execution failures under the original task instructions are also considered. It achieves effective execution-failure detection at a low FPR: no baseline attains a higher mean execution-failure TPR at an equal or lower mean FPR.

Relative to spatial monitoring alone, incorporating action uncertainty improves execution-failure detection and further increases pooled TPR and balanced accuracy. The extended monitor achieves the highest values on these two pooled metrics among all evaluated methods, while maintaining a similarly low FPR and a comparable mean TWA. Thus, the extension strengthens detection of execution failures while preserving the overall advantages of spatial monitoring.

Table 4: Runtime monitoring on $\pi_{0.5}$. Exec. TPR is computed on execution failures under the original task instructions.

Method	FPR ↓	TPR ↑	Bal. Acc. ↑	TWA ↑	Det. Time ↓	Exec. TPR ↑
CALM	$0.086_{\pm 0.006}$	$0.970_{\pm 0.012}$	$0.942_{\pm 0.003}$	$0.894_{\pm 0.004}$	$0.099_{\pm 0.006}$	$0.640_{\pm 0.126}$
CALM_{ext}	$0.082_{\pm 0.013}$	$0.981_{\pm 0.008}$	$0.950_{\pm 0.010}$	$0.888_{\pm 0.006}$	$0.125_{\pm 0.012}$	$0.777_{\pm 0.085}$
SAFE	$0.133_{\pm 0.029}$	$0.862_{\pm 0.132}$	$0.864_{\pm 0.057}$	$0.787_{\pm 0.051}$	$0.176_{\pm 0.118}$	$0.792_{\pm 0.081}$
Hide-and-Seek	$0.121_{\pm 0.057}$	$0.668_{\pm 0.064}$	$0.773_{\pm 0.060}$	$0.676_{\pm 0.036}$	$0.288_{\pm 0.058}$	$0.795_{\pm 0.073}$
ActProbe	$0.083_{\pm 0.064}$	$0.328_{\pm 0.218}$	$0.623_{\pm 0.083}$	$0.559_{\pm 0.038}$	$0.359_{\pm 0.069}$	$0.536_{\pm 0.467}$
CoVer	$0.113_{\pm 0.059}$	$0.827_{\pm 0.135}$	$0.857_{\pm 0.039}$	$0.732_{\pm 0.029}$	$0.306_{\pm 0.026}$	$0.458_{\pm 0.160}$
GPLA	$0.102_{\pm 0.015}$	$0.804_{\pm 0.079}$	$0.851_{\pm 0.039}$	$0.820_{\pm 0.043}$	$0.079_{\pm 0.018}$	$0.408_{\pm 0.044}$
VLA-FAIL	$0.071_{\pm 0.061}$	$0.330_{\pm 0.158}$	$0.630_{\pm 0.051}$	$0.544_{\pm 0.019}$	$0.547_{\pm 0.081}$	$0.467_{\pm 0.238}$
RND-OE	$0.145_{\pm 0.009}$	$0.198_{\pm 0.028}$	$0.527_{\pm 0.017}$	$0.502_{\pm 0.015}$	$0.248_{\pm 0.014}$	$0.161_{\pm 0.047}$
logpZO	$0.121_{\pm 0.006}$	$0.248_{\pm 0.035}$	$0.564_{\pm 0.015}$	$0.524_{\pm 0.012}$	$0.322_{\pm 0.023}$	$0.161_{\pm 0.047}$
Rel. Mahalanobis	$0.117_{\pm 0.081}$	$0.612_{\pm 0.084}$	$0.747_{\pm 0.029}$	$0.614_{\pm 0.024}$	$0.440_{\pm 0.042}$	$0.839_{\pm 0.047}$
Deep k-NN	$0.125_{\pm 0.031}$	$0.397_{\pm 0.085}$	$0.636_{\pm 0.028}$	$0.539_{\pm 0.007}$	$0.486_{\pm 0.004}$	$0.747_{\pm 0.057}$
FOREWARN	$0.063_{\pm 0.048}$	$0.006_{\pm 0.006}$	$0.471_{\pm 0.022}$	$0.469_{\pm 0.024}$	–	$0.045_{\pm 0.039}$
SEAL	$0.074_{\pm 0.037}$	$0.127_{\pm 0.003}$	$0.526_{\pm 0.017}$	$0.515_{\pm 0.017}$	$0.177_{\pm 0.008}$	$0.045_{\pm 0.039}$
I-FailSense	$0.043_{\pm 0.024}$	$0.063_{\pm 0.009}$	$0.510_{\pm 0.009}$	$0.498_{\pm 0.009}$	$0.386_{\pm 0.039}$	$0.000_{\pm 0.000}$
CLIP	$0.086_{\pm 0.018}$	$0.015_{\pm 0.006}$	$0.464_{\pm 0.005}$	$0.457_{\pm 0.009}$	–	$0.089_{\pm 0.031}$
ProbeAct	$0.219_{\pm 0.047}$	$0.189_{\pm 0.080}$	$0.485_{\pm 0.022}$	$0.435_{\pm 0.008}$	$0.529_{\pm 0.036}$	$0.420_{\pm 0.316}$
FIPER	$0.332_{\pm 0.066}$	$0.614_{\pm 0.054}$	$0.641_{\pm 0.012}$	$0.523_{\pm 0.023}$	$0.385_{\pm 0.020}$	$0.729_{\pm 0.051}$
ACE	$0.187_{\pm 0.071}$	$0.496_{\pm 0.057}$	$0.655_{\pm 0.008}$	$0.537_{\pm 0.018}$	$0.476_{\pm 0.013}$	$0.661_{\pm 0.047}$
DiffDAgger	$0.156_{\pm 0.076}$	$0.468_{\pm 0.051}$	$0.656_{\pm 0.017}$	$0.554_{\pm 0.026}$	$0.435_{\pm 0.006}$	$0.616_{\pm 0.085}$
TC	$0.203_{\pm 0.032}$	$0.401_{\pm 0.086}$	$0.599_{\pm 0.032}$	$0.494_{\pm 0.016}$	$0.526_{\pm 0.019}$	$0.702_{\pm 0.055}$
RND-A	$0.223_{\pm 0.041}$	$0.205_{\pm 0.056}$	$0.491_{\pm 0.010}$	$0.435_{\pm 0.006}$	$0.559_{\pm 0.035}$	$0.229_{\pm 0.051}$
Similarity	$0.152_{\pm 0.039}$	$0.101_{\pm 0.030}$	$0.474_{\pm 0.005}$	$0.456_{\pm 0.012}$	$0.367_{\pm 0.048}$	$0.205_{\pm 0.015}$
FM-accel	$0.168_{\pm 0.049}$	$0.548_{\pm 0.064}$	$0.690_{\pm 0.010}$	$0.580_{\pm 0.011}$	$0.400_{\pm 0.020}$	$0.682_{\pm 0.036}$
UPS	$0.848_{\pm 0.019}$	$0.749_{\pm 0.010}$	$0.451_{\pm 0.006}$	$0.376_{\pm 0.009}$	$0.201_{\pm 0.041}$	$0.932_{\pm 0.005}$
KnowNo	$0.891_{\pm 0.028}$	$0.636_{\pm 0.012}$	$0.373_{\pm 0.010}$	$0.200_{\pm 0.012}$	$0.544_{\pm 0.006}$	$0.932_{\pm 0.005}$

Table 5: Fusion of spatial support deficits and action uncertainty.

Rule	TPR	TNR	Bal. Acc.	TWA	Det. Time	FPR	Exec. TPR	Goal repl.	Absent
Sum	$0.981_{\pm 0.008}$	$0.918_{\pm 0.013}$	$0.950_{\pm 0.010}$	$0.888_{\pm 0.006}$	$0.125_{\pm 0.012}$	$0.082_{\pm 0.013}$	$0.777_{\pm 0.085}$	$1.000_{\pm 0.000}$	$1.000_{\pm 0.000}$
Conjunction	$0.513_{\pm 0.026}$	$0.977_{\pm 0.000}$	$0.745_{\pm 0.013}$	$0.614_{\pm 0.003}$	$0.510_{\pm 0.014}$	$0.023_{\pm 0.000}$	$0.500_{\pm 0.000}$	$0.514_{\pm 0.025}$	$0.517_{\pm 0.076}$
Disjunction	$0.974_{\pm 0.003}$	$0.883_{\pm 0.052}$	$0.928_{\pm 0.028}$	$0.857_{\pm 0.038}$	$0.146_{\pm 0.033}$	$0.117_{\pm 0.052}$	$0.682_{\pm 0.036}$	$1.000_{\pm 0.000}$	$1.000_{\pm 0.000}$
Fisher combination	$0.981_{\pm 0.018}$	$0.930_{\pm 0.050}$	$0.956_{\pm 0.025}$	$0.854_{\pm 0.018}$	$0.208_{\pm 0.013}$	$0.070_{\pm 0.050}$	$0.783_{\pm 0.194}$	$1.000_{\pm 0.000}$	$1.000_{\pm 0.000}$
Rank-normalized sum	$0.927_{\pm 0.033}$	$0.829_{\pm 0.070}$	$0.878_{\pm 0.019}$	$0.685_{\pm 0.020}$	$0.417_{\pm 0.028}$	$0.171_{\pm 0.070}$	$0.827_{\pm 0.175}$	$0.928_{\pm 0.022}$	$1.000_{\pm 0.000}$

D.3 Fusion and Readout Sensitivity

Combining spatial and action evidence. Table 5 evaluates how spatial support and action uncertainty should be combined. Requiring both channels to trigger substantially reduces detection of goal replacement and absent targets, because weak action-side evidence can suppress the instruction-sensitive spatial signal. Disjunctive fusion avoids this failure mode but detects fewer execution failures and incurs a higher false-positive rate than additive fusion. By combining the two signals before calibration, additive fusion preserves instruction-related detection while allowing action uncertainty to contribute evidence for execution failures. These results support forming a single monitoring statistic rather than requiring separate channel-level decisions.

Conjunction and disjunction apply AND and OR, respectively, to separately calibrated channel decisions. The per-channel calibration level is α for conjunction and $\alpha/2$ for disjunction.

E Experimental Details

Policies and evaluation conditions. We monitor three frozen policies on the ten LIBERO-10 tasks (Liu et al., 2023): $\pi_{0.5}$ (`pi05_libero`, Physical Intelligence et al., 2025), π_0 (`pi0_libero`, Black et al., 2025) and OpenVLA (`openvla-7b-finetuned-libero-10`, Kim et al., 2024). Rollouts use the official initial states, environment seed, warm-up procedure and task-dependent step limits. The two flow-matching policies replan every five executed steps, and baselines that consume a predicted chunk receive the chunk of the policy they monitor; OpenVLA emits one action per

step. *Execution failures* are unsuccessful rollouts under the original task instruction, labelled by the environment's success flag.

Dataset and splits. For each policy we collect 570 rollouts with the same composition: 300 under the official task instructions, 150 under goal replacements, 50 with absent targets and 70 with ambiguous instructions. The split is the same for all three policies: per seed, half of each task's official rollouts together with three absent-target rollouts per task form the training partition, and the unsuccessful official rollouts in it are removed, so CALM is trained only on successful rollouts under the official instructions plus the absent-target examples that supply the abstention label. Goal-replacement and ambiguous-instruction rollouts are never used for training or calibration.

Readout inputs and architecture. The patch grid is the frozen policy's image-token output, kept at its native 16×16 resolution ($HW = 256$ cells) with no pooling; $\mathbf{h}_t$ is the hidden state of the last layer at the final prompt token. The shared adapter is a two-layer spectral-normalised MLP $(256, 128)$ concatenated with a fixed random projection of its input (64 dimensions); the abstention feature uses the same construction on $\mathbf{h}_t$ alone (128, 64 plus 32), giving $d_0 = 96$. The Bayesian last layer uses a dense covariance and a zero-mean Gaussian prior, with the KL weight set to $1/N$ for N training frames. The endpoint label is the cell occupied by the end effector ten executed steps after the replan.

Training and calibration. The adapter and the Bayesian last layer are trained with Adam, learning rate 10^{-3}, batch size 128 and 3,000 updates; prediction averages $S = 64$ posterior samples and reads the 3×3 neighbourhood of Eq. 7. The threshold is the order statistic of Eq. 10 on held-out successful official rollouts at $\alpha = 0.15$. Results are reported over seeds 0, 1 and 2.

Baselines. We compare against runtime failure predictors and uncertainty scores: RND-OE (Römer et al., 2025), RND-A (He et al., 2026), embedding similarity (Agia et al., 2025), logpZO (Xu et al., 2025), temporal consistency (TC) (Agia et al., 2025), ACE (Römer et al., 2025), DiffDAgger (Lee et al., 2025), and FIPER (Römer et al., 2025); representation-based OOD detectors: relative Mahalanobis distance (Ren et al., 2021) and deep k-nearest-neighbor distance (Sun et al., 2022); and the flow-based uncertainty monitor FM-accel (Rao et al., 2026). We also evaluate SAFE (Gu et al., 2025), ActProbe (Huang et al., 2026), Hide-and-Seek (Park et al., 2026), VLA-FAIL (Seligmann et al., 2026), CoVer (Kwok et al., 2026), GPLA (Wulff et al., 2026), ProbeAct (Zhang et al., 2026a), KnowNo (Ren et al., 2023), UPS (Yuan et al., 2026), FOREWARN (Wu et al., 2025), SEAL (Wu et al., 2026), I-FailSense (Grislain et al., 2026), and CLIP similarity (Radford et al., 2021). Baselines follow their respective training and calibration procedures. For each policy and seed, all methods are evaluated on the same test episodes. We also evaluate logistic regression and MLP classifiers for direct intervention prediction, with and without failure labels (Appendix H.2).

E.1 Metric Definitions

Let $\mathcal{E}_+$ and $\mathcal{E}_-$ denote the positive and negative test-episode index sets, and let A_e indicate whether episode e triggers ASK. Then

$$\mathrm{TPR} = \frac{1}{|\mathcal{E}_+|} \sum_{e \in \mathcal{E}_+} A_e, \qquad \mathrm{FPR} = \frac{1}{|\mathcal{E}_-|} \sum_{e \in \mathcal{E}_-} A_e, \tag{24}$$

and balanced accuracy is $\mathrm{BalAcc} = (\mathrm{TPR} + 1 - \mathrm{FPR})/2$. Condition-specific TPR restricts $\mathcal{E}_+$ to the relevant instruction condition or to execution failures under the original instructions.

Let $\rho_e \in [0, 1]$ denote the normalized time of the first ASK for a detected positive episode. Detection time is averaged over detected positives:

$$\mathrm{DetTime} = \frac{\sum_{e \in \mathcal{E}_+ : A_e = 1} \rho_e}{\sum_{e \in \mathcal{E}_+} A_e}. \tag{25}$$

Following Römer et al. (2025); Park et al. (2026), we report balanced time-weighted accuracy, abbreviated as TWA:

$$\mathrm{TWA} = \frac{1}{2} \left[\frac{1}{|\mathcal{E}_+|} \sum_{e \in \mathcal{E}_+ : A_e = 1} (1 - \rho_e) + (1 - \mathrm{FPR}) \right]. \tag{26}$$

The first term rewards early detection, with undetected positives contributing zero; the second rewards uninterrupted successful execution.

Table 6: **Matched-state goal replacement.** Observations, proprioceptive states, and denoising noise are fixed across instructions. Success and failure refer to outcomes under the original instruction. Rates and monitoring signals are mean ± standard deviation over three seeds; decision-change counts are listed by seed.

	Episode ASK rate		Changed episodes (seed 1/2/3)	
Monitor / episodes	Original	Replacement	No ASK → ASK	ASK → No ASK
CALM, all ($n = 300$)	0.131 ± 0.010	0.991 ± 0.010	$257/259/258$	$0/0/0$
successful ($n = 271$)	0.064 ± 0.009	0.990 ± 0.011	$250/254/249$	$0/0/0$
execution failures ($n = 29$)	0.759 ± 0.069	1.000 ± 0.000	$7/5/9$	$0/0/0$
Flow Instability only ($n = 300$)	0.267 ± 0.034	0.330 ± 0.048	$48/37/43$	$24/22/25$

Mean per-replan signal	Original	Replacement
Spatial support deficit $q_{e,t}$	0.014 ± 0.002	0.552 ± 0.050
Abstention probability $\bar{p}_{e,t,0}$	0.007 ± 0.003	0.548 ± 0.050
Flow Instability $\chi_{e,t}$ (Rao et al., 2026)	0.131 ± 0.018	0.142 ± 0.020

Action chunks compared	Mean change
Consecutive replans, original instruction	0.133
Same state, original vs. replacement instruction	0.050

Table 7: Spatial-monitor ablations on $\pi_{0.5}$ at $\alpha = 0.15$. The abstention-only variant uses the predicted abstention probability as its monitoring score.

Variant	FPR ↓	TPR ↑	Bal. Acc. ↑	TWA ↑	Det. Time ↓
CALM	0.086 ± 0.006	0.970 ± 0.012	0.942 ± 0.003	0.894 ± 0.004	0.099 ± 0.006
Deterministic last layer	0.094 ± 0.030	0.959 ± 0.012	0.933 ± 0.020	0.883 ± 0.011	0.104 ± 0.021
Without patch grid	0.137 ± 0.053	0.979 ± 0.009	0.921 ± 0.022	0.810 ± 0.009	0.227 ± 0.029
Abstention probability only	0.090 ± 0.048	0.957 ± 0.008	0.934 ± 0.023	0.900 ± 0.016	0.071 ± 0.016
Without abstention output	0.153 ± 0.049	0.981 ± 0.003	0.914 ± 0.024	0.842 ± 0.015	0.148 ± 0.028

All metrics are computed separately for each seed and then summarized by their mean and standard deviation. Detection time is undefined when no positive is detected.

F Additional ablation Experiments

F.1 Spatial Readout Ablations

Tables 6 and 7 present the matched-state instruction analysis and spatial readout ablations discussed in the main text.

Table 8 shows that both variants produce high support deficits and near-complete detection across the three instruction–execution mismatch conditions. The spatial readout thus remains informative without an explicit abstention output, which reduces unnecessary interruptions.

G Additional Policy Backbones

G.1 A Second Flow-Based Policy

On the frozen π_0 policy (Black et al., 2025), CALM achieves the highest TPR, balanced accuracy, and TWA among methods whose mean FPR is at most $\alpha = 0.15$ (Table 9). Together with the OpenVLA results, this supports the effectiveness of spatial monitoring across the tested flow-based and autoregressive policies.

Table 8: Per-condition detection rates and mean support deficits with and without the abstention output, on $\pi_{0.5}$ at $\alpha = 0.15$. For each variant, the first row reports TPR for the first four conditions and FPR on successful episodes under official instructions in the last column. The second row reports the mean per-frame support deficit $q_{e,t}$ for the corresponding condition.

Variant	Goal repl. ↑	Absent ↑	Ambiguous ↑	Exec. TPR ↑	FPR ↓
CALM	$1.000_{\pm 0.000}$	$1.000_{\pm 0.000}$	$1.000_{\pm 0.000}$	$0.640_{\pm 0.126}$	$0.086_{\pm 0.006}$
Mean $q_{e,t}$	$0.670_{\pm 0.042}$	$0.961_{\pm 0.012}$	$0.622_{\pm 0.063}$	$0.071_{\pm 0.006}$	$0.014_{\pm 0.000}$
Without abstention output	$1.000_{\pm 0.000}$	$1.000_{\pm 0.000}$	$0.995_{\pm 0.008}$	$0.771_{\pm 0.051}$	$0.153_{\pm 0.049}$
Mean $q_{e,t}$	$0.667_{\pm 0.055}$	$0.920_{\pm 0.012}$	$0.647_{\pm 0.056}$	$0.144_{\pm 0.044}$	$0.042_{\pm 0.011}$

Table 9: Instruction–execution mismatch detection on the π_0.

Method	FPR ↓	TPR ↑	Bal. Acc. ↑	TWA ↑	Det. Time ↓
Within FPR budget (≤ 0.15)					
CALM	$0.134_{\pm 0.037}$	$\mathbf{0.970}_{\pm 0.005}$	$\mathbf{0.918}_{\pm 0.016}$	$\mathbf{0.840}_{\pm 0.017}$	$\underline{0.162}_{\pm 0.017}$
SAFE	$0.129_{\pm 0.034}$	$0.886_{\pm 0.054}$	$0.878_{\pm 0.024}$	$0.793_{\pm 0.022}$	$0.192_{\pm 0.031}$
ActProbe	$0.096_{\pm 0.054}$	$0.794_{\pm 0.015}$	$0.849_{\pm 0.025}$	$0.728_{\pm 0.021}$	$0.305_{\pm 0.022}$
GPLA	$0.115_{\pm 0.011}$	$0.783_{\pm 0.062}$	$0.834_{\pm 0.026}$	$\underline{0.807}_{\pm 0.015}$	$\mathbf{0.068}_{\pm 0.023}$
VLA-FAIL	$0.041_{\pm 0.033}$	$0.457_{\pm 0.098}$	$0.708_{\pm 0.034}$	$0.581_{\pm 0.013}$	$0.561_{\pm 0.026}$
ACE	$0.147_{\pm 0.101}$	$0.307_{\pm 0.052}$	$0.580_{\pm 0.025}$	$0.508_{\pm 0.030}$	$0.476_{\pm 0.055}$
DiffDAgger	$0.094_{\pm 0.061}$	$0.606_{\pm 0.040}$	$0.756_{\pm 0.024}$	$0.616_{\pm 0.018}$	$0.465_{\pm 0.036}$
TC	$0.102_{\pm 0.054}$	$0.677_{\pm 0.085}$	$0.787_{\pm 0.016}$	$0.657_{\pm 0.014}$	$0.387_{\pm 0.033}$
Similarity	$0.130_{\pm 0.065}$	$0.684_{\pm 0.105}$	$0.777_{\pm 0.046}$	$0.644_{\pm 0.050}$	$0.395_{\pm 0.053}$
FM-accel	$0.131_{\pm 0.045}$	$0.878_{\pm 0.022}$	$0.873_{\pm 0.013}$	$0.753_{\pm 0.008}$	$0.275_{\pm 0.019}$
Rel. Mahalanobis	$0.087_{\pm 0.074}$	$\underline{0.907}_{\pm 0.060}$	$\underline{0.910}_{\pm 0.014}$	$0.740_{\pm 0.017}$	$0.378_{\pm 0.049}$
Deep k-NN	$0.095_{\pm 0.039}$	$0.880_{\pm 0.035}$	$0.893_{\pm 0.019}$	$0.721_{\pm 0.023}$	$0.391_{\pm 0.028}$
FOREWARN	$0.105_{\pm 0.008}$	$0.049_{\pm 0.007}$	$0.472_{\pm 0.007}$	$0.462_{\pm 0.007}$	$0.422_{\pm 0.071}$
SEAL	$0.056_{\pm 0.026}$	$0.138_{\pm 0.004}$	$0.541_{\pm 0.014}$	$0.528_{\pm 0.012}$	$0.185_{\pm 0.030}$
I-FailSense	$0.020_{\pm 0.009}$	$0.107_{\pm 0.006}$	$0.543_{\pm 0.007}$	$0.529_{\pm 0.006}$	$0.257_{\pm 0.013}$
CLIP	$0.099_{\pm 0.017}$	$0.016_{\pm 0.005}$	$0.458_{\pm 0.009}$	$0.451_{\pm 0.008}$	–
Above FPR budget (> 0.15)					
Hide-and-Seek	$0.240_{\pm 0.128}$	$0.898_{\pm 0.065}$	$0.829_{\pm 0.040}$	$0.765_{\pm 0.037}$	$0.143_{\pm 0.014}$
CoVer	$0.180_{\pm 0.023}$	$0.954_{\pm 0.014}$	$0.887_{\pm 0.007}$	$0.821_{\pm 0.018}$	$0.139_{\pm 0.024}$
ProbeAct	$0.353_{\pm 0.109}$	$0.380_{\pm 0.029}$	$0.513_{\pm 0.067}$	$0.410_{\pm 0.077}$	$0.547_{\pm 0.087}$
FIPER	$0.289_{\pm 0.030}$	$0.831_{\pm 0.072}$	$0.771_{\pm 0.050}$	$0.649_{\pm 0.060}$	$0.296_{\pm 0.051}$
RND-OE	$0.177_{\pm 0.075}$	$0.793_{\pm 0.090}$	$0.808_{\pm 0.009}$	$0.686_{\pm 0.020}$	$0.312_{\pm 0.065}$
RND-A	$0.199_{\pm 0.098}$	$0.538_{\pm 0.095}$	$0.669_{\pm 0.007}$	$0.545_{\pm 0.017}$	$0.467_{\pm 0.027}$
logpZO	$0.207_{\pm 0.082}$	$0.357_{\pm 0.057}$	$0.575_{\pm 0.069}$	$0.491_{\pm 0.053}$	$0.467_{\pm 0.040}$
UPS	$1.000_{\pm 0.000}$	$1.000_{\pm 0.000}$	$0.500_{\pm 0.000}$	$0.500_{\pm 0.000}$	$0.000_{\pm 0.000}$
KnowNo	$1.000_{\pm 0.000}$	$0.981_{\pm 0.003}$	$0.490_{\pm 0.002}$	$0.295_{\pm 0.006}$	$0.398_{\pm 0.010}$

H Additional Calibration Analysis

H.1 Sensitivity to the Interruption Level

Table 10 varies α using the same trained spatial predictor and successful calibration episodes. Larger α improves detection coverage, including execution failures, and advances assistance requests, but increases FPR. Mean TWA remains similar from $\alpha = 0.05$ to 0.15 and decreases at $\alpha = 0.20$, where additional interruptions outweigh the gain in timely detection.

H.2 Ablations of the Intervention Rule

Table 11 evaluates how alternative intervention rules affect detection and unnecessary interruptions for the spatial monitor.

Table 10: Interruption-level sensitivity for the spatial monitor on $\pi_{0.5}$.

α	FPR ↓	TPR ↑	Bal. Acc. ↑	TWA ↑	Det. Time ↓	Exec. TPR ↑
0.05	0.028 ± 0.018	0.948 ± 0.013	0.960 ± 0.010	0.888 ± 0.004	0.153 ± 0.027	0.429 ± 0.071
0.10	0.063 ± 0.019	0.959 ± 0.003	0.948 ± 0.009	0.893 ± 0.008	0.116 ± 0.008	0.497 ± 0.067
0.15	0.086 ± 0.006	0.970 ± 0.012	0.942 ± 0.003	0.894 ± 0.004	0.099 ± 0.006	0.640 ± 0.126
0.20	0.164 ± 0.041	0.980 ± 0.011	0.908 ± 0.019	0.866 ± 0.016	0.086 ± 0.007	0.756 ± 0.119

Table 11: Intervention-rule ablations for the spatial monitor on $\pi_{0.5}$.

Rule	TPR ↑	FPR ↓	Bal. Acc. ↑	TWA ↑	Det. Time ↓	Exec. TPR ↑
Threshold calibration						
CALM	$0.970_{\pm 0.012}$	$0.086_{\pm 0.006}$	$0.942_{\pm 0.003}$	$0.894_{\pm 0.004}$	$0.099_{\pm 0.006}$	$0.640_{\pm 0.126}$
Functional band on $q_{*,t}$	$0.993_{\pm 0.003}$	$0.308_{\pm 0.061}$	$0.842_{\pm 0.032}$	$0.806_{\pm 0.023}$	$0.072_{\pm 0.021}$	$0.908_{\pm 0.044}$
Functional band on $Z_{*,t}$	$0.972_{\pm 0.015}$	$0.144_{\pm 0.063}$	$0.914_{\pm 0.024}$	$0.881_{\pm 0.017}$	$0.067_{\pm 0.016}$	$0.667_{\pm 0.149}$
PAC tolerance	$0.957_{\pm 0.003}$	$0.047_{\pm 0.021}$	$0.955_{\pm 0.011}$	$0.894_{\pm 0.012}$	$0.127_{\pm 0.010}$	$0.476_{\pm 0.041}$
Posterior-based decisions						
Bayesian quadrature	$0.957_{\pm 0.003}$	$0.039_{\pm 0.030}$	$0.959_{\pm 0.015}$	$0.891_{\pm 0.013}$	$0.141_{\pm 0.035}$	$0.476_{\pm 0.041}$
VBLL credible level	$0.998_{\pm 0.003}$	$0.860_{\pm 0.019}$	$0.569_{\pm 0.009}$	$0.546_{\pm 0.013}$	$0.046_{\pm 0.009}$	$0.976_{\pm 0.041}$
Direct intervention prediction						
Logistic classifier, no failure labels	$0.946_{\pm 0.011}$	$0.097_{\pm 0.037}$	$0.924_{\pm 0.014}$	$0.888_{\pm 0.012}$	$0.078_{\pm 0.006}$	$0.345_{\pm 0.090}$
MLP classifier, no failure labels	$0.953_{\pm 0.006}$	$0.140_{\pm 0.083}$	$0.906_{\pm 0.039}$	$0.867_{\pm 0.037}$	$0.084_{\pm 0.005}$	$0.432_{\pm 0.072}$
Logistic classifier, with failure labels	$1.000_{\pm 0.000}$	$0.687_{\pm 0.021}$	$0.656_{\pm 0.011}$	$0.635_{\pm 0.004}$	$0.043_{\pm 0.014}$	$1.000_{\pm 0.000}$
MLP classifier, with failure labels	$0.996_{\pm 0.003}$	$0.890_{\pm 0.093}$	$0.553_{\pm 0.048}$	$0.539_{\pm 0.043}$	$0.029_{\pm 0.012}$	$0.952_{\pm 0.041}$

We compare threshold calibration, posterior-based decisions, and direct intervention classifiers trained with or without failure labels.

Table 11 compares alternative intervention rules for the spatial monitor.

Threshold calibration. Functional conformal bands trigger earlier, but incur substantially more unnecessary interruptions and lower TWA than our episode-level rule. PAC tolerance further reduces FPR, but detects fewer execution failures and intervenes later, with comparable TWA.

Posterior-based decisions. Bayesian quadrature similarly reduces FPR, but detects fewer execution failures and triggers later. In contrast, the VBLL credible-level rule attains high detection rates by interrupting a large fraction of successful episodes, resulting in substantially lower TWA.

Direct intervention prediction. Without failure labels, both classifiers achieve lower TPR and execution-failure TPR than CALM while generally using more of the false-positive budget. Providing failure labels increases detection, but at the cost of frequent unnecessary interruptions and substantially lower TWA.

I Deployment Cost

We measure per-replan latency on an RTX 6000 Ada at batch size one. Table 12 reports the shared policy inference latency, additional monitoring latency (UQ), and the ratio of monitoring latency to policy latency.

Compared with methods requiring additional vision–language inference, CALM achieves stronger detection at lower computational cost (Tables 1 and 12). It also improves detection over computationally lighter action-based monitors while keeping its overhead small relative to policy inference.

Table 12: Per-replan computation costs on an RTX 6000 Ada at batch size one. Policy is the shared action-generation cost; UQ is the additional monitoring computation. The ratio is UQ divided by the policy cost.

Method	Policy (ms)	UQ (ms)	× policy
Rel. Mahalanobis	84.1	0.13	0.002
Hide-and-Seek	84.1	0.16	0.002
ActProbe	84.1	0.17	0.002
FM-accel	84.1	0.29	0.003
Similarity	84.1	0.44	0.005
SAFE	84.1	0.47	0.006
RND-OE	84.1	0.49	0.006
RND-A	84.1	0.53	0.006
ProbeAct	84.1	0.63	0.007
Deep k-NN	84.1	0.76	0.009
logpZO	84.1	1.42	0.017
CALM	84.1	1.80	0.021
CoVer	84.1	7.20	0.086
GPLA	84.1	7.25	0.086
CLIP	84.1	14.37	0.171
DiffDAgger	84.1	48.58	0.578
VLA-FAIL	84.1	55.98	0.666
I-FailSense	84.1	245.91	2.925
UPS	84.1	282.04	3.353
KnowNo	84.1	286.56	3.409
TC	84.1	713.71	8.489
ACE	84.1	714.38	8.497
FIPER	84.1	714.69	8.501
FOREWARN	84.1	1103.57	13.122
SEAL	84.1	1767.68	21.019