CERTVLA: CERTIFIED DEFENSE AGAINST PHYSICAL VISUAL ATTACKS FOR VISION-LANGUAGE-ACTION MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Vision-Language-Action (VLA) policies are vulnerable to localized physical perturbations, yet existing certified patch defenses target discrete labels and cannot directly certify continuous, temporally correlated actions. We introduce CertVLA, a certified defense for closed-loop VLA control under bounded patch and texture attacks. CertVLA proposes a calibrated region of behaviorally consistent actions, while deterministic covering masks ensure that at least one checked prediction is attack-free. Specifically, CertVLA normalizes action disagreement by the benign variation of each mask pair and accepts a single-mask anchor only when it remains consistent under every second mask. It then calibrates the resulting max-min-max episode score to provide finite-sample clean coverage. Conjoining query-level decisions extends the action certificate to the complete closed-loop rollout. Furthermore, we prove that against any adaptive attacker satisfying the bounded-support threat model, every rollout certified by CertVLA executes only action chunks consistent with attack-erased clean predictions. Under dual-mask rollout correctness, this consistency certificate further guarantees task success. The certificate is independent of attack content, shape, generation method, location, temporal variation, or physical transformation. Experiments in simulation and the real world demonstrate the empirical and certified effectiveness of CertVLA against patch attacks, with **video demos** available in the **Supplementary Material**.

1 INTRODUCTION

Vision-Language-Action (VLA) models connect visual perception and language instructions directly to robot control, enabling general-purpose policies for diverse manipulation tasks (Zitkovich et al., 2023; Octo Model Team et al., 2024; Black et al., 2024; Physical Intelligence et al., 2025). This tight perception-action coupling also creates a physical attack surface: a localized patch in the scene or an adversarial texture on an object can persist across viewpoints and corrupt an entire closed-loop trajectory. Recent attacks have demonstrated such failures across tasks, VLA architectures, and sim-to-real settings (Wang et al., 2025; Lu et al., 2026; Chen et al., 2026). Existing empirical defenses improve VLA robustness through robust training, decoupled robustness learning, safety-constrained optimization, and noise-filtering modules (Guo et al., 2025; Xie et al., 2026; Fu et al., 2026), while patch-specific defenses in vision typically localize, mask, or restore suspicious regions (Tarchoun et al., 2023; Xu et al., 2023; Jing et al., 2024; Yankelev et al., 2026). For safety-critical deployment, however, empirical robustness alone is insufficient, as resistance to tested attacks does not establish robustness to all attacks within the threat model (Cohen et al., 2019; Levine & Feizi, 2020b; Xiang et al., 2022). A certified defense should guarantee protection against any attack within the bounded attacked region, regardless of its optimization strategy, adaptivity, or attack form.

Existing certified patch defenses provide a natural starting point for this goal. Prior work has developed deterministic ablation, restricted receptive fields, and masking-based approaches to certify discrete predictions against bounded patch attacks (Levine & Feizi, 2020a; Xiang et al., 2021). Extending this paradigm to VLAs, however, requires addressing fundamental differences between discrete perception and embodied control. *First*, VLAs typically produce continuous action chunks, for which exact agreement among masked outputs is generally too restrictive and a measurable notion of action consistency is needed. *Second*, although covering-based masking (Lyu et al., 2026;

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

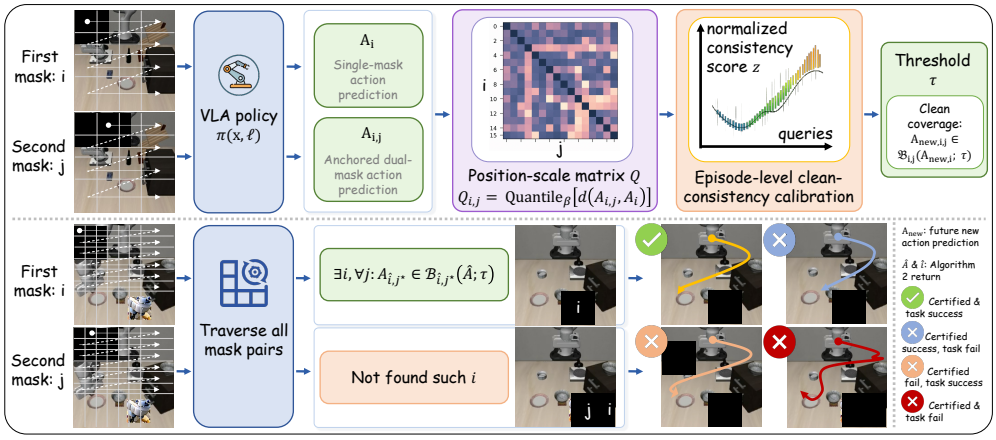


Figure 1: CertVLA framework. *Top:* clean single- and dual-mask predictions estimate the position-scale matrix $\mathbf{Q}$ and calibrate threshold τ. *Bottom:* at each query, CertVLA returns a single-mask anchor only if it is consistent with every second-mask prediction, then composes query flags into an episode certificate distinct from empirical task success.

Xiang et al., 2022) can handle an unknown patch location, its discrete recovery criterion does not directly apply to continuous actions. The defender must instead determine whether a recovered action remains within a calibrated region of an attack-erased reference without knowing which mask removes the attack. *Third*, VLA actions are executed in closed loop. A certificate at one policy query only constrains the current action chunk, while its execution changes the state and observations encountered at subsequent queries. Certification must therefore extend from individual predictions to the sequence of decisions made along the actual rollout. Hence, a research question arises naturally:

> *How can we certify continuous VLA actions against unknown bounded physical perturbations and extend the guarantee from individual policy queries to closed-loop task execution?*

In response, we introduce *CertVLA*, a model-agnostic certified defense for continuous VLA control under bounded patch and texture attacks. CertVLA uses an $\mathcal{R}$-covering masks (Xiang et al., 2022) so that, for any admissible support, at least one evaluated mask completely erases the corrupted region. However, because the attack location is unknown, this existential guarantee alone cannot certify any particular single-mask action. Therefore, CertVLA treats each single-mask prediction as an anchor and checks it against every second-mask prediction, including one guaranteed to erase the attack. Because different mask positions induce different benign action changes, we normalize each anchored dual-mask disagreement by a position-specific clean scale and calibrate a shared tolerance with held-out clean episodes. The resulting max-min-max score exactly represents the required $\forall$-query–$\exists$-anchor–$\forall$-second-mask decision. At deployment, an early-stopped recovery rule rejects a row as soon as it cannot pass and returns a certified anchor only if all second-mask checks pass.

This construction yields a deliberately layered guarantee. Whenever CertVLA certifies a query, the returned action chunk lies in a calibrated clean-consistency region of a prediction for which the attack has been erased. The statement holds for arbitrary patch content and adaptive attack generation, provided that the attack support can be fully enclosed by a fixed-size square region. Conjoining query-level flags extends the result to the complete executed rollout, even when the admissible patch location changes over time. Under an explicit dual-mask rollout-correctness condition, the same certificate further implies terminal task success.

Our contributions are summarized as: **1)** We formulate certified defense for continuous closed-loop VLA control and introduce a calibrated action-consistency region beyond exact label agreement. **2)** We develop an anchored dual-mask defense that couples position-aware normalization with a covering-based recovery rule for certifiable action selection. **3)** We prove query- and episode-level consistency guarantees against any bounded-support adaptive attacker, and isolate the precise additional condition under which this observable certificate guarantees task completion. **4)** Experiments across multiple VLAs, attack types, and deployments show the broad applicability of CertVLA.

2 Related Work

VLA models. VLA policies map vision and language directly to robot actions. Early scaling were established by RT-1, PaLM-E, and RT-2 (Brohan et al., 2022; Driess et al., 2023; Zitkovich et al., 2023). RoboFlamingo adapts open VLMs through imitation learning, whereas Open X-Embodiment and Octo exploit heterogeneous cross-robot data (Li et al., 2023; Open X-Embodiment Collaboration et al., 2023; Octo Model Team et al., 2024). OpenVLA provides an open 7B policy and OpenVLA-OFT improves action generation through parallel decoding and action chunking (Kim et al., 2025b;a). π_0 and $\pi_{0.5}$ policies use flow matching and heterogeneous co-training for dexterous, open-world control (Black et al., 2024; Physical Intelligence et al., 2025). Recent VLAs continue to improve generalization, cross-embodiment control, and closed-loop manipulation through larger-scale training, experience-driven learning, and active perception (Intelligence et al., 2025; Liu et al., 2026), while certifiable robustness to physical adversarial perturbations remains largely unexplored.

Adversarial attacks & defense in robotics. Printable patches remain effective across physical transformations (Brown et al., 2017). RoboticAttack demonstrates untargeted, trajectory-targeted, and physical attacks on VLA control, while UPA-RFAS transfers patches across architectures, tasks, viewpoints, and sim-to-real settings (Wang et al., 2025; Lu et al., 2026). Tex3D further turns 3D object textures into persistent attacks (Chen et al., 2026). Recent defenses improve empirical stability to visual or multimodal perturbations (Guo et al., 2025; Xie et al., 2026; Fu et al., 2026), but their observed robustness does not offer a per-rollout guarantee against adaptive attacks. CertVLA instead certifies any visible corruption whose projected support satisfies the bounded-region threat model.

Certified defenses against patch attacks. Patch certificates use interval bounds or deterministic ablation (Chiang et al., 2020; Levine & Feizi, 2020a), restricted receptive fields and masking (Xiang et al., 2021; Xiang & Mittal, 2021), or occlusion consistency (McCoyd et al., 2020; Xiang et al., 2022). ScaleCert improves high-resolution certification, PatchCURE exposes tunable robustness–utility–efficiency trade-offs, and CertMask reduces redundant mask evaluations (Han et al., 2021; Xiang et al., 2024; Lyu et al., 2026). PatchDEMUX extends single-label certificates to multi-label prediction (Jacob et al., 2025). Nevertheless, these guarantees rely on discrete label or detection decisions. Exact agreement is unsuitable for continuous, temporally correlated action chunks, and an isolated prediction certificate does not cover states reached later in a rollout. CertVLA addresses both gaps through calibrated anchored consistency and episode-level composition.

3 Preliminaries

VLA Policy and Threat Model. Let $x_t = (x_t^{\text{ext}}, s_t)$ contain the external-camera image and proprioceptive state at policy query index t, respectively, and let ℓ be a language instruction. A VLA policy π produces an action chunk $A_t = \pi(x_t, \ell) \in \mathcal{A} \subseteq \mathbb{R}^{H \times D}$ of horizon H and action dimension D. The controller executes only the first $h \leq H$ actions before next query. The adversary may arbitrarily replace pixels on a support δ_t, which may vary over time and take any shape, provided that it is fully covered by some region $\Omega_t \in \mathcal{R}(P)$. Here, Ω_t may also vary with t, and $\mathcal{R}(P)$ contains every axis-aligned region no larger than $P \times P$ pixels. For external camera view,

$$x_t^{\delta} = (1 - \mathbf{1}_{\delta_t}) \odot x_t + \mathbf{1}_{\delta_t} \odot \Delta_t, \qquad \delta_t \subseteq \Omega_t,\ \Omega_t \in \mathcal{R}(P), \tag{1}$$

where $\mathbf{1}_{\delta_t}$ indicates the actual attacked support, Δ_t is arbitrary adversarial pixel content, and $\odot$ denotes the Hadamard product. Thus, pixels in $\Omega_t \setminus \delta_t$ remain unchanged from x_t. We make no assumptions about the attack content, shape, generation method, location, temporal variation, or physical transformation. Certification only requires that the attacked support projected into defended view be fully coverable by a region in $\mathcal{R}(P)$.

Covering Masks and Action Predictions. The attacked support is unknown at inference time, so no single predetermined mask can guarantee removal. Following PatchCleanser Xiang et al. (2022), we enumerate K deterministic masks, where index $i \in \{1, \ldots, K\}$ and operator M_i overwrites a fixed region with a benign value. The covering property guarantees that, for any admissible support, at least one mask completely covers it, independent of its shape or appearance.

Definition 1 ($\mathcal{R}$-covering family) *A deterministic family $\mathcal{M} = \{M_1, \ldots, M_K\}$ is $\mathcal{R}$-covering if, for every $\Omega \in \mathcal{R}(P)$, some mask $M_{j^\star}$ overwrites all pixels in Ω.*

For a query (x, ℓ), we define the unmasked, single-mask, and anchored dual-mask predictions

$$A_0 = \pi(x, \ell),\ A_i = \pi(M_i(x), \ell),\ A_{i,j} = \pi(M_j(M_i(x)), \ell). \tag{2}$$

For an attacked input x^Ω, superscript Ω denotes the analogous predictions $A_i^\Omega = \pi(M_i(x^\Omega), \ell)$ and $A_{i,j}^\Omega = \pi(M_j(M_i(x^\Omega)), \ell)$. This notation separates the attacked quantities observed at deployment from their clean counterparts. The single-mask prediction A_i, rather than every pair of dual-mask predictions, will serve as the anchor of a candidate recovery row.

4 METHODOLOGY

Overview. Fig. 1 summarizes CertVLA's calibration and deployment stages. On clean episodes, CertVLA normalizes mask-dependent action variation and calibrates an episode-level threshold. At deployment, it returns a single-mask anchor only when the anchor is consistent with every second-mask prediction, and certifies the rollout only if every policy query passes. The covering property then ties each certified action to a clean prediction from which the attack is erased.

4.1 CONTINUOUS-ACTION CONSISTENCY SCORE

Unlike discrete labels (Xiang et al., 2022; Lyu et al., 2026), behaviorally equivalent action chunks need not match exactly. We therefore measure normalized action deviation and account for the distinct benign scale of each ordered mask pair.

Action Distance. Let $[a_u^-, a_u^+]$ be the valid range of action coordinate $u \in \{1, \dots, D\}$ and define

$$d(A, B) = \frac{1}{hD} \sum_{r=1}^{h} \sum_{u=1}^{D} \frac{|A_{r,u} - B_{r,u}|}{\eta_u(B_{r,u})},\ \eta_u(b) = \max\{|a_u^+ - b|, |b - a_u^-|, \epsilon\}. \tag{3}$$

$A, B \in \mathbb{R}^{H \times D}$ are the dual-mask chunk and single-mask anchor, r indexes the h executed actions, and u indexes action coordinates. The denominator normalizes each coordinate by its valid range, with $\epsilon > 0$ ensuring numerical stability. Since it is evaluated at the anchor B, $d(A, B)$ is directional.

Position-scale Matrix. Mask pairs occlude different scene content and therefore have different clean deviation scales. To normalize these pair-specific deviations while preserving their positional dependence, we define a position-scale matrix $\mathbf{Q}$. Let e index an episode, $q \in \{1, \dots, N_e^{\text{qry}}\}$ a policy query, N_e^{qry} the number of queries, and $i, j \in \{1, \dots, K\}$ an ordered mask pair. Let $A_{e,q,i}$ and $A_{e,q,i,j}$ be the corresponding single- and dual-mask predictions. From clean scale-calibration episodes $\mathcal{C}_{\text{scale}}$, we collect

$$\mathcal{D}_{i,j}^{\text{scale}} = \{d(A_{e,q,i,j}, A_{e,q,i}) : e \in \mathcal{C}_{\text{scale}},\quad q = 1, \dots, N_e^{\text{qry}}\}, \tag{4}$$

and set its reference scale to the fixed β-quantile ($\beta \in (0, 1)$ is chosen before row calibration),

$$Q_{i,j} = \widehat{\text{Quantile}}_\beta\big(\mathcal{D}_{i,j}^{\text{scale}}\big). \tag{5}$$

Therefore, the matrix $\mathbf{Q} = [Q_{i,j}] \in \mathbb{R}_{\geq 0}^{K \times K}$ removes pair-specific scale.

4.2 POSITION-AWARE CLEAN-CONSISTENCY CALIBRATION

The local scales $\mathbf{Q}$ make mask pairs comparable but do not set a joint acceptance threshold. We therefore calibrate an episode-level normalized radius τ, using disjoint scale and row splits to preserve the exchangeability for conformal calibration. We define normalized consistency score

$$z_{e,q,i,j} = d(A_{e,q,i,j}, A_{e,q,i})/(Q_{i,j} + \epsilon), \tag{6}$$

where ϵ handles zero empirical variation. Thus $z_{e,q,i,j}$ measures deviation relative to the clean behavior of the same ordered pair, enabling a shared threshold across positions.

Definition 2 (Pair-specific clean-consistency region) *For an ordered mask pair (i, j), anchor action $B \in \mathcal{A}$, and tolerance $\tau \geq 0$, define*

$$\mathcal{B}_{i,j}(B; \tau) := \{A \in \mathcal{A} : d(A, B) \leq \tau(Q_{i,j} + \epsilon)\}. \tag{7}$$

We call A (i, j, τ)-consistent with B when $A \in \mathcal{B}_{i,j}(B; \tau)$. Here, $\tau(Q_{i,j} + \epsilon)$ is the raw tolerance for pair (i, j).

Algorithm 1 Position-Aware Episode Calibration

Require: Scale episodes $\mathcal{C}_{\text{scale}}$; row episodes $\mathcal{C}_{\text{row}}$; masks $\mathcal{M}$; quantile β; clean-episode miscoverage level α; ϵ

Ensure: Position-scale matrix $\mathbf{Q} = [Q_{i,j}]$ and threshold τ

 # Stage I: position-scale estimation
for $i = 1$ to K **do**
  for $j = 1$ to K **do**
    $\mathcal{D}^{\text{scale}}_{i,j} \leftarrow \{d(A_{e,q,i,j}, A_{e,q,i}) : e \in \mathcal{C}_{\text{scale}}, q = 1, \ldots, N^{\text{qry}}_e\}$; $Q_{i,j} \leftarrow \widehat{\text{Quantile}}_{\beta}(\mathcal{D}^{\text{scale}}_{i,j})$
  end for
end for
 # Stage II: episode-level joint calibration
for all $e \in \mathcal{C}_{\text{row}}$ **do**
  $z_{e,q,i,j} \leftarrow d(A_{e,q,i,j}, A_{e,q,i})/(Q_{i,j} + \epsilon)$ for all q, i, j; $S_e \leftarrow \max_q \min_i \max_j z_{e,q,i,j}$
end for
$\tau \leftarrow \widehat{Q}^{\text{conf}}_{1-\alpha}(\{S_e\})$
return $(\mathbf{Q}, \tau)$

The following lemma bridges the normalized pairwise score used for calibration and the action-space consistency region used by the subsequent certificate.

Lemma 1 (Score-region equivalence) *For every episode e, policy query q, ordered mask pair (i, j), and tolerance $\tau \geq 0$,*

$$z_{e,q,i,j} \leq \tau \quad \Longleftrightarrow \quad A_{e,q,i,j} \in \mathcal{B}_{i,j}(A_{e,q,i}; \tau). \tag{8}$$

The equivalence follows from $Q_{i,j} + \epsilon > 0$; see *Proof A.1.* Alg. 1 partitions clean, unattacked episodes into $\mathcal{C}_{\text{scale}}$ and $\mathcal{C}_{\text{row}}$. Stage I estimates each ordered-pair scale $Q_{i,j}$ by Equation 5 and freezes $\mathbf{Q}$ before Stage II. Stage II calibrates the event dictated by $\mathcal{R}$-covering: at each query, some row must be consistent for every second mask, including the unknown mask that erases the patch:

$$\mathsf{Pass}_e(\tau) := \big[\forall q\ \exists i\ \forall j : A_{e,q,i,j} \in \mathcal{B}_{i,j}(A_{e,q,i}; \tau)\big]. \tag{9}$$

Its scalar score is

$$S_e = \max_q \min_i \max_j z_{e,q,i,j}. \tag{10}$$

Here $\max_j$, $\min_i$, and $\max_q$ implement $\forall j$, $\exists i$, and $\forall q$, respectively. Consequently, S_e is the minimum threshold required for every policy query in episode e to admit at least one row whose normalized disagreement remains within the threshold for all second-mask positions. The next lemma lifts this pairwise interpretation to the episode level, showing that a scalar score exactly represents the dual-mask requirement that every query admit an anchor consistent across all second masks.

Lemma 2 (Score-quantifier equivalence) *For all episode e and threshold $\tau \geq 0$, we have (proof in App. A.2.):*

$$S_e \leq \tau \Longleftrightarrow \mathsf{Pass}_e(\tau) \Longleftrightarrow \forall q\ \exists i\ \forall j :\ z_{e,q,i,j} \leq \tau. \tag{11}$$

Episode-level conformal calibration. Though $\mathbf{Q}$ normalizes the location-dependent disagreement of individual mask pairs, the aggregated score S_e can still vary across clean episodes due to variations in initial states, object configurations, executed trajectories, and policy-query sequences. Thus, we calibrate the global acceptance threshold at the episode level rather than fixing it heuristically.

Let $\alpha \in (0, 1)$ denote the target clean-episode miscoverage rate. Given $n = |\mathcal{C}_{\text{row}}|$ calibration episodes with ordered scores $S_{(1)} \leq \cdots \leq S_{(n)}$, we set

$$\tau = S_{(k)}, \qquad k = \lceil (n+1)(1-\alpha) \rceil, \qquad \text{s.t. } k \leq n. \tag{12}$$

Under the standard exchangeability assumption between the calibration episodes and a future clean episode, this conformal quantile provides finite-sample marginal coverage $\Pr(S_{\text{new}} \leq \tau) \geq 1 - \alpha$. Equivalently, a new clean episode is rejected because its consistency score exceeds τ with probability at most α. Thus, α controls the allowed clean-episode rejection rate, while τ is the corresponding data-dependent tolerance for episode-level action consistency. A smaller α yields a larger τ, reducing clean rejection at the cost of a looser consistency certificate. At deployment, fixed $Q_{i,j}$ supplies

Algorithm 2 DUALMASKRECOVER

Require: (x, ℓ); VLA π; masks $\mathcal{M}$; $\mathbf{Q}$; τ; ϵ
Ensure: Action $\widehat{A}$, selected row $\hat{i}$, flag c

$R_{\text{best}} \leftarrow +\infty$; $A_{\text{best}} \leftarrow \pi(x, \ell)$; $\hat{i} \leftarrow 0$
for $i = 1$ to K **do**
  $A_i \leftarrow \pi(M_i(x), \ell)$; $R_i \leftarrow 0$
  for $j = 1$ to K **do**
    $A_{i,j} \leftarrow \pi(M_j(M_i(x)), \ell)$; $R_i \leftarrow \max\left\{R_i, \frac{d(A_{i,j}, A_i)}{Q_{i,j}+\epsilon}\right\}$; **if** $R_i > \tau$ **then break**
  end for
  if $R_i \leq \tau$ **then return** $(A_i, i, \textbf{true})$; **if** $R_i < R_{\text{best}}$ **then** $(R_{\text{best}}, A_{\text{best}}, \hat{i}) \leftarrow (R_i, A_i, i)$
end for
return $(A_{\text{best}}, \hat{i}, \textbf{false})$

Algorithm 3 Defended Rollout Evaluation

Require: Environment $\mathcal{E}$; instruction ℓ; VLA π; masks $\mathcal{M}$; $\mathbf{Q}$; τ; ϵ; horizon T
Ensure: Task outcome S_{def} and episode flag C

Reset $\mathcal{E}$; $C \leftarrow$ **true**; $S_{\text{def}} \leftarrow$ **false**
for $t = 0$ to $T - 1$ **do**
  if a new policy query is required **then**
    $(\widetilde{A}_t, \hat{i}_t, c_t) \leftarrow$ DUALMASKRECOVER$(x_t, \ell, \pi, \mathcal{M}, \mathbf{Q}, \tau, \epsilon)$; $C \leftarrow C \wedge c_t$
  end if
  Execute the next action from $\widetilde{A}_t$ in $\mathcal{E}$; **If** task success **then** $S_{\text{def}} \leftarrow$ **true**; **break**
end for
return (S_{def}, C)

the pair-specific unit and fixed τ thresholds the joint score. Together, Lemmas 1 and 2 give the exact action-region and episode-level interpretations. Finally, the following lemma shows that conformal calibration turns this episode-level acceptance rule into a finite-sample guarantee on the acceptance probability of a future clean episode.

Lemma 3 (Finite-sample clean coverage) *If the row-calibration episodes and a future new episode are exchangeable, Alg. 1 satisfies* $\Pr[S_{\text{new}} \leq \tau] \geq 1-\alpha$. *Equivalently, with probability at least* $1-\alpha$, *the future episode satisfies* $\forall q\, \exists i\, \forall j : A_{\text{new},q,i,j} \in \mathcal{B}_{i,j}(A_{\text{new},q,i}; \tau)$. *Proof in App. A.3.*

4.3 ANCHORED DUAL-MASK RECOVERY

The patch location is unknown, so each first-mask prediction serves as an anchor and is checked against every second mask. By $\mathcal{R}$-covering, at least one check erases the patch.

Lemma 4 (Attack erasure) *Let* x^{Ω} *differ from* x *only on* $\Omega \in \mathcal{R}(P)$, *and let each mask overwrite its support with fixed values. If* $\mathcal{M}$ *is* $\mathcal{R}$*-covering, some* $M_{j^\star}$ *covers* Ω *and, for every* $M_i \in \mathcal{M}$,

$$M_{j^\star}(M_i(x^{\Omega})) = M_{j^\star}(M_i(x)). \tag{13}$$

Alg. 2 updates a running row score as second masks are evaluated. Because this score is nondecreasing, a row can be safely discarded as soon as it exceeds τ. A row that survives all K checks has the complete score $R_i = \max_{1 \leq j \leq K} \frac{d(A_{i,j}, A_i)}{Q_{i,j}+\epsilon}$. The algorithm immediately returns the first complete row satisfying $R_i \leq \tau$. Hence $c = \text{true}$ only after verifying every second mask, whereas early stopping merely avoids unnecessary evaluations of a rejected row. If no row passes, it returns the anchor with the smallest observed partial score as an explicitly uncertified fallback.

Theorem 1 (Attack-erased consistency) *Let* x^{Ω} *be any attacked input with* $\Omega \in \mathcal{R}(P)$. *If Alg. 2 returns* $(\widehat{A}, \hat{i}, \text{true})$, *then* $\widehat{A} = A_{\hat{i}}^{\Omega}$ *and there exists a clean dual-masked prediction* $A_{\hat{i},j^\star}$ *such that*

$$A_{\hat{i},j^\star} \in \mathcal{B}_{\hat{i},j^\star}(\widehat{A}; \tau) \iff d(A_{\hat{i},j^\star}, \widehat{A}) \leq \tau(Q_{\hat{i},j^\star} + \epsilon) \leq \tau(Q_{\hat{i}}^{\max} + \epsilon), \tag{14}$$

where $M_{j^\star}$ covers Ω and $Q_{\hat{i}}^{\max} = \max_j Q_{\hat{i},j}$. The bound is independent of the patch content, shape, attack-generation method, and physical transformation, provided that the attack support $\Omega \in \mathcal{R}(P)$ satisfies the certified size bound and can therefore be fully covered by at least one mask in $\mathcal{M}$.

See *App. A.5* for proof. Notably, Theorem 1 provides a safety-relevant robustness guarantee for action outputs: under any attack satisfying the bounded-support threat model, each certified action chunk is provably consistent, within the calibrated tolerance, with an attack-erased clean prediction. For practicality, the patch-erased prediction should preserve task-relevant behavior. *App. C* empirically validates this property across VLA models under random single- and dual-masking.

4.4 Closed-Loop Episode Certificate

Action consistency certification alone does not imply terminal task success. Thus, we further state the required closed-loop condition for continuous actions.

Definition 3 (Dual-mask rollout correctness) *Let $\mathcal{T}_{\mathrm{qry}}$ denote the policy-query times of a defended rollout, and let $\widetilde{A}_t$ be the recovered action chunk executed at query time t. A task, policy, and environment satisfy* dual-mask rollout correctness *if every defended rollout for which*

$$\forall t \in \mathcal{T}_{\mathrm{qry}}, \quad \exists (i_t, j_t): \quad A^{\mathrm{clean}}_{t,i_t,j_t} \in \mathcal{B}_{i_t,j_t}(\widetilde{A}_t; \tau) \tag{15}$$

reaches the task-success set. Here $A^{\mathrm{clean}}_{t,i_t,j_t}$ denotes the patch-erased dual-mask prediction evaluated at the same reached state as $\widetilde{A}_t$.

At each $t \in \mathcal{T}_{\mathrm{qry}}$, Alg. 3 invokes Alg. 2, executes from the recovered chunk $\widetilde{A}_t$, and accumulates $C = \bigwedge_{t \in \mathcal{T}_{\mathrm{qry}}} c_t$. Because Equation 3 jointly scores its action chunk, aggregation is per query. The outputs separate defended success S_{def}, episode certification C, and certified success $S_{\mathrm{def}} \wedge C$, and any uncertified fallback makes C false.

Theorem 2 (Closed-loop consistency and conditional task certificate) *Suppose every query's patch support belongs to $\mathcal{R}(P)$ and $\mathcal{M}$ is $\mathcal{R}$-covering. If Alg. 3 returns $C = \mathrm{true}$, then at every $t \in \mathcal{T}_{\mathrm{qry}}$ there exist indices $(i_t, j_t^\star)$ and an attack-erased clean prediction satisfying $A^{\mathrm{clean}}_{t,i_t,j_t^\star} \in \mathcal{B}_{i_t,j_t^\star}(\widetilde{A}_t; \tau)$. This action-consistency conclusion remains valid if the admissible patch location changes between queries. Moreover, if the task, policy, and environment additionally satisfy Definition 3, then $C = \mathrm{true}$ implies $S_{\mathrm{def}} = \mathrm{true}$ for every admissible patch sequence.*

Theorem 2 provides a layered certificate tailored to closed-loop continuous control: its first clause verifies an attacker-independent action-consistency property, while its conditional second clause invokes the explicit rollout-correctness condition only for the stronger task-success claim. This separation keeps the guarantee logically transparent while allowing S_{def}, C, and $S_{\mathrm{def}} \wedge C$ to be evaluated without conflating empirical success with certification (See App. E).

5 Experiments

Dataset & threat models. We use LIBERO (Liu et al., 2023), a simulated manipulation benchmark, and evaluate its Spatial, Object, Goal, and Long suites with 10 tasks for each, including spatial relations, object interaction, goal-conditioned control, and multi-stage execution. We report results for 4 VLA policies: OpenVLA (Kim et al., 2025b), OpenVLA-OFT (Kim et al., 2025a), π_0 (Black et al., 2024), and $\pi_{0.5}$ (Physical Intelligence et al., 2025), across all suites. Attacks are physical patches (Wang et al., 2025; Lu et al., 2026) and adversarial object textures (Chen et al., 2026). In both cases, the visible image-plane corruption must satisfy the bounded-support threat model.

Evaluation metrics & details. For each model, task suite, and attack setting, we evaluate N closed-loop episodes per suite. *Defense* reports the defended task success rate, $N^{-1}\sum_{e=1}^{N} \mathbb{I}[S_e^{\mathrm{def}} = 1]$, where S_e^{def} denotes task success under defense. *Certified* reports certified task success rate $N^{-1}\sum_{e=1}^{N} \mathbb{I}[S_e^{\mathrm{def}} = 1 \wedge C_e = \mathrm{true}]$, where C_e means all policy queries in episode e pass the consistency test. In plain terms, Certified task success is the fraction of attacked rollouts in which

Table 1: Defense and certified task success rates (%) against physical patch and adversarial object texture attacks on LIBERO. Average denotes the mean performance across the four LIBERO suites. † denotes our dual-mask fine-tuned model, and fine-tuning details are provided in Appendix B.

Attack Type ↓	Model ↓	LIBERO-Spatial		LIBERO-Object		LIBERO-Goal		LIBERO-Long		Average	
		Def.	Cert.	Def.	Cert.	Def.	Cert.	Def.	Cert.	Def.	Cert.
Patch attack (Wang et al., 2025; Lu et al., 2026)	OpenVLA†	96	96	96	64	96	94	88	76	94.00	82.50
	OpenVLA-OFT	98	98	94	94	98	98	86	86	94.00	94.00
	π_0	82	36	92	84	86	61	74	66	83.50	61.75
	$\pi_{0.5}$	96	72	97	89	84	68	86	77	90.75	76.50
Texture attack (Chen et al., 2026)	OpenVLA†	84	82	77	30	83	83	37	30	70.25	56.25
	OpenVLA-OFT	88	88	94	92	90	90	40	37	78.00	76.75
	π_0	88	38	92	80	82	62	74	70	84.00	62.50
	$\pi_{0.5}$	100	92	94	88	94	88	90	86	94.50	88.50

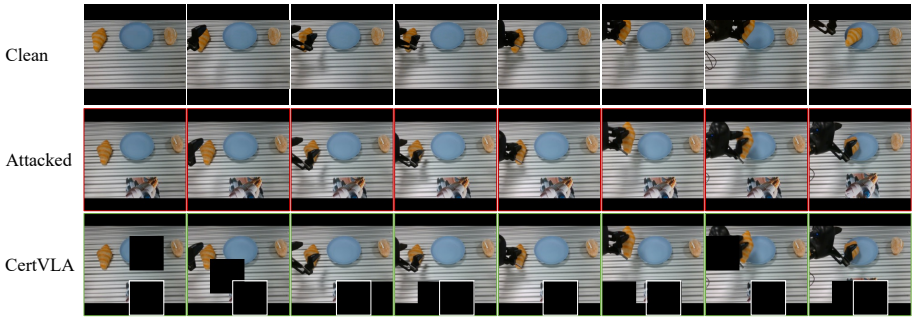


Figure 2: Real-robot rollouts across successive policy queries. The bottom row is diagnostic: it visualizes the dual-mask candidate with the smallest normalized score z at each query. The white outline identifies the anchor mask of the accepted row.

the robot both completes the task with CertVLA and obtains a valid certificate that every action chunk executed during the rollout satisfies the calibrated clean-consistency bound with an attack-erased clean prediction. We set $\beta = 0.95$, $\alpha = 0.5$, and $\epsilon = 10^{-8}$ for all main results. See *App. D* for more experimental details.

Physical Experiment Setting. For real-world experiments, we employ a dual-arm Piper robotic platform. Each arm is controlled using a 7-dimensional end-effector action. We use one arm to implement the pick-and-place task. Visual observations are provided by two RGB cameras: one Intel RealSense D435 camera offering a third-person view, which we use, and one Intel RealSense D405 camera mounted on the wrist. Our policy uses these synchronized RGB observations together with the arm proprioceptive state. We use standardized objects and workspace layouts for systematic quantitative evaluation, with the task repeated for 10 independent trials.

5.1 Main Results

Defense against patch & texture attacks in simulation. In Tab. 1, OpenVLA-OFT obtains 94% average Defense and Certified success, while OpenVLA† reaches 94% Defense and 82.5% Certified success. The larger gaps for the generative policies show that some successful rollouts contain a query outside the calibrated region, confirming the value of reporting empirical recovery and certification separately. Defense against texture attack shows transfer beyond planar patches. $\pi_{0.5}$ performs best, averaging 94.5% Defense and 88.5% Certified success; π_0 reaches 84% and 62.5%, respectively. OpenVLA variants are less stable on the Long suite. The result matches our support-based threat model: certification depends on covering the visible corruption, not on how its texture was generated. See App. E for more certified metrics analysis and App. G for more interpretation.

Defense against physical patch attack in the real world. *Quantitative results.* Tab. 2 shows that the patch reduces $\pi_{0.5}$ success from 90% to 40%. CertVLA reaches 60%, a 20-point gain that recovers 40% of the attack-induced loss. Its 30% Certified success means that half of the successfully defended trials pass at every query. The remain-

Table 2: Results for physical patch attacks on the real robot.

Model	Clean	Attack	Defense	Certified
$\pi_{0.5}$	90	40	60	30

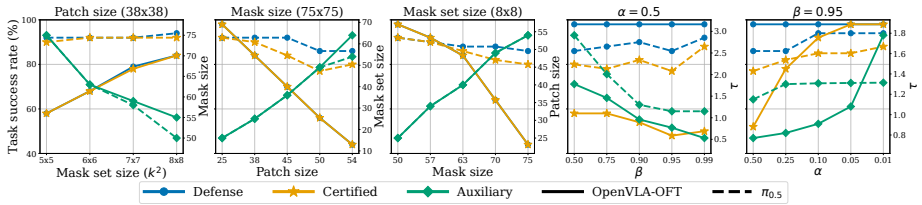


Figure 3: Sensitivity to mask-set size, certified patch size, mask size, pair-scale quantile β, and conformal level α (left to right). Blue and orange report Defense and Certified task success; green reports the corresponding auxiliary quantity on the right axis.

ing Defense-Certified gap identifies successful rollouts with at least one query outside the calibrated region. It reflects a stricter certificate under physical environment. *Qualitative analysis.* Fig. 2 separates execution from diagnostic checks. Its third row shows only a reference dual-mask candidate. The one with minimum z-score at each query is illustrated, which does not select the action. Alg. 2 instead finds an anchor i whose entire row satisfies $z_{i,j} \leq \tau$ $(\exists i \, \forall j)$. The white box marks this anchor, and A_i is predicted from an input masking only its pixels. The other black region visualizes one second-mask check and is absent during execution. Thus, dual masks establish the certificate while the executed single-mask input retains more task-relevant content.

5.2 Discussion

Covering, utility, and calibration trade-offs. Fig. 3 analyzes the principal design parameters from left to right. Let P, M, and s denote the patch side length, mask side length, and mask stride. The mask family is $\mathcal{R}$-covering only if $P \leq M - s + 1$ so only configurations satisfying this constraint are considered. Across these valid configurations, $\pi_{0.5}$ maintains a consistently high Defense rate, whereas OpenVLA-OFT is more sensitive to the masking configuration. With $P = 38$ fixed, increasing the mask set provides denser coverage and permits smaller masks, reducing visual occlusion at the cost of more mask evaluations. This substantially improves OpenVLA-OFT Defense, while$\pi_{0.5}$ remains consistently high. Certified success also increases for both models, indicating that finer coverage benefits both task utility and certifiable consistency. With $M = 75$ fixed, larger certified patches require denser mask placement and a larger mask set to maintain $\mathcal{R}$-coverage. But both Defense and Certified rates decrease as P grows, showing that certifying a larger attack region places greater demands on the masked policy. With the mask set fixed at $8{\times}8$, increasing M enlarges the occluded region while keeping the number of candidate masks unchanged. Both Defense and Certified rates decrease as M grows, indicating that larger masks remove more task-relevant visual information and make the recovered actions less reliable. The degradation is stronger for OpenVLA-OFT, whereas $\pi_{0.5}$ remains more stable, showing greater tolerance to increased occlusion under the same covering density.

Calibration is also policy dependent. Increasing β rescales $Q_{i,j}$ and changes normalized τ non-monotonically, and we set it to 0.95 to mitigate the influence of extreme values while preventing an overly large τ and an excessively loose certificate. Reducing α raises acceptance for both policies but loosens the consistency region (Lemma 3). See App. F for efficiency analysis.

Interpreting the certification gap. Defense success measures if the recovered policy completes a task, whereas Certified success additionally requires every executed query to satisfy the calibrated consistency test. A rollout may therefore succeed yet remain uncertified because of one ambiguous observation. Small gaps indicate stable masked actions throughout an episode, while larger gaps expose queries that may gain from improved masked-model utility or calibration.

6 Conclusion

CertVLA provides a certified defense for continuous closed-loop VLA control by combining position-normalized action consistency and episode-level calibration. Our analysis establishes query- and rollout-level guarantees that relate certified actions to attack-erased predictions under bounded-support attacks, with task success further guaranteed when dual-mask rollout correctness holds. Experiments demonstrate both empirical defense effectiveness and certifiable robustness across VLA models and physical attacks.

AI USE STATEMENT

Generative AI tools were used to assist with code implementation, feedback on experimental design and analysis, literature search, and the polishing of manuscript text. The research problem and core method were developed by the authors. All AI-assisted code was reviewed and tested, and all reported results, claims, citations, and figures were manually verified by the authors. Generative AI was not used to generate experimental datasets or determine the final scientific conclusions. We take responsibility for the final content of this work, including text, claims, code, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work investigates the robustness and safety of vision-language-action (VLA) models against physical visual attacks. Our goal is to characterize vulnerabilities arising from adversarial patches and textures and to develop certified defenses and evaluation protocols that support safer robotic deployment. Experiments use a publicly available simulation benchmark and controlled real-robot environments with standardized objects. All physical attack experiments are conducted on systems and equipment under the researchers' control and do not target third-party robots or infrastructure. The proposed certificates apply under the explicitly stated threat model and assumptions. We are not aware of any additional ethical concerns or conflicts of interest associated with this work.

REPRODUCIBILITY STATEMENT

We provide the methodological and experimental details necessary for reproducing our results in the main paper and appendix. These include descriptions of the proposed method, experimental settings, datasets, evaluation protocols, and relevant implementation details.

REFERENCES

Kevin Black, Noah Brown, Danny Driess, et al. π_0: A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 2024. URL https://arxiv.org/abs/2410.24164.

Anthony Brohan, Noah Brown, Justice Carbajal, et al. RT-1: Robotics transformer for real-world control at scale. *arXiv preprint arXiv:2212.06817*, 2022.

Tom B. Brown, Dandelion Mané, Aurko Roy, Martin Abadi, and Justin Gilmer. Adversarial patch. *arXiv preprint arXiv:1712.09665*, 2017. URL https://arxiv.org/abs/1712.09665.

Jiawei Chen, Simin Huang, Jiawei Du, Shuaihang Chen, Yu Tian, Mingjie Wei, Chao Yu, and Zhaoxia Yin. Tex3D: Objects as attack surfaces via adversarial 3d textures for vision-language-action models. *arXiv preprint arXiv:2604.01618*, 2026. URL https://arxiv.org/abs/2604.01618.

Ping-yeh Chiang, Renkun Ni, Ahmed Abdelkader, Chen Zhu, Christoph Studor, and Tom Goldstein. Certified defenses for adversarial patches. In *International Conference on Learning Representations*, 2020. URL https://arxiv.org/abs/2003.06693.

Jeremy Cohen, Elan Rosenfeld, and Zico Kolter. Certified adversarial robustness via randomized smoothing. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 1310–1320. PMLR, 2019.

Danny Driess, Fei Xia, Mehdi S. M. Sajjadi, et al. PaLM-E: An embodied multimodal language model. In *International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, 2023.

Yiyang Fu, Chubin Zhang, Shukai Gong, Yufan Deng, Kaiwei Sun, Qiyang Min, Qibin Hou, Yansong Tang, Jianan Wang, and Daquan Zhou. StableVLA: Towards robust vision-language-action models without extra data. In *International Conference on Machine Learning*, 2026.

Jianing Guo, Zhenhong Wu, Chang Tu, Yiyao Ma, Xiangqi Kong, Zhiqian Liu, Jiaming Ji, Shuning Zhang, Yuanpei Chen, Kai Chen, Xianglong Liu, Qi Dou, Yaodong Yang, Huijie Zhao, Weifeng Lv, and Simin Li. On robustness of vision-language-action model against multi-modal perturbations. *arXiv preprint arXiv:2510.00037*, 2025.

Husheng Han, Kaidi Xu, Xing Hu, Xiaobing Chen, Ling Liang, Zidong Du, Qi Guo, Yanzhi Wang, and Yunji Chen. ScaleCert: Scalable certified defense against adversarial patches with sparse superficial layers. In *Advances in Neural Information Processing Systems*, volume 34, 2021.

Physical Intelligence, Ali Amin, Raichelle Aniceto, Ashwin Balakrishna, Kevin Black, Ken Conley, Grace Connors, James Darpinian, Karan Dhabalia, Jared DiCarlo, et al. $\pi^*_{0.6}$: a vla that learns from experience. *arXiv preprint arXiv:2511.14759*, 2025.

Dennis Jacob, Chong Xiang, and Prateek Mittal. PatchDEMUX: A certifiably robust framework for multi-label classifiers against adversarial patches. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9944–9953, 2025.

Lihua Jing, Rui Wang, Wenqi Ren, Xin Dong, and Cong Zou. PAD: Patch-agnostic defense against adversarial patch attacks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24472–24481, 2024.

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

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, et al. Openvla: An open-source vision-language-action model. In *Conference on Robot Learning*, volume 270, pp. 2679–2713. PMLR, 2025b. URL https://proceedings.mlr.press/v270/kim25c.html.

Alexander Levine and Soheil Feizi. (de)randomized smoothing for certifiable defense against patch attacks. In *Advances in Neural Information Processing Systems*, volume 33, 2020a. URL https://arxiv.org/abs/2002.10733.

Alexander Levine and Soheil Feizi. Robustness certificates for sparse adversarial attacks by randomized ablation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 4585–4593, 2020b. doi: 10.1609/aaai.v34i04.5888.

Xinghang Li, Minghuan Liu, Hanbo Zhang, et al. Vision-language foundation models as effective robot imitators. *arXiv preprint arXiv:2311.01378*, 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 *Advances in Neural Information Processing Systems*, volume 36, 2023.

Zhenyang Liu, Yongchong Gu, Yikai Wang, Xiangyang Xue, and Yanwei Fu. ActiveVLA: Injecting active perception into vision-language-action models for precise 3d robotic manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8141–8151, 2026.

Hui Lu, Yi Yu, Yiming Yang, Chenyu Yi, Qixin Zhang, Bingquan Shen, Alex C. Kot, and Xudong Jiang. When robots obey the patch: Universal transferable patch attacks on vision-language-action models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22867–22878, 2026.

Xuntao Lyu, Ching-Chi Lin, Abdullah Al Arafat, Georg von der Brüggen, Jian-Jia Chen, and Zhishan Guo. CertMask: Certifiable defense against adversarial patches via theoretically optimal mask coverage. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 7735–7743, 2026. doi: 10.1609/aaai.v40i10.37716.

Michael McCoyd, Won Park, Steven Chen, Neil Shah, Ryan Roggenkemper, Minjune Hwang, Jason Xinyu Liu, and David Wagner. Minority reports defense: Defending against adversarial patches. *arXiv preprint arXiv:2004.13799*, 2020.

Octo Model Team, Dibya Ghosh, et al. Octo: An open-source generalist robot policy. In *Robotics: Science and Systems*, 2024. URL https://arxiv.org/abs/2405.12213.

Open X-Embodiment Collaboration, Abby O'Neill, Abdul Rehman, et al. Open x-embodiment: Robotic learning datasets and RT-X models. *arXiv preprint arXiv:2310.08864*, 2023.

Physical Intelligence, Kevin Black, Noah Brown, et al. $\pi_{0.5}$: A vision-language-action model with open-world generalization. *arXiv preprint arXiv:2504.16054*, 2025. URL https://arxiv.org/abs/2504.16054.

Bilel Tarchoun, Anouar Ben Khalifa, Mohamed Ali Mahjoub, Nael Abu-Ghazaleh, and Ihsen Alouani. Jedi: Entropy-based localization and removal of adversarial patches. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4087–4095, 2023.

Taowen Wang, Dongfang Liu, James Chenhao Liang, Wenhao Yang, Qifan Wang, Cheng Han, Jiebo Luo, and Ruixiang Tang. Exploring the adversarial vulnerabilities of vision-language-action models in robotics. In *IEEE/CVF International Conference on Computer Vision*, pp. 6948–6958, 2025. URL https://arxiv.org/abs/2411.13587.

Chong Xiang and Prateek Mittal. PatchGuard++: Efficient provable attack detection against adversarial patches. *arXiv preprint arXiv:2104.12609*, 2021.

Chong Xiang, Arjun Nitin Bhagoji, Vikash Sehwag, and Prateek Mittal. PatchGuard: A provably robust defense against adversarial patches via small receptive fields and masking. In *30th USENIX Security Symposium*, pp. 2237–2254. USENIX Association, 2021.

Chong Xiang, Saeed Mahloujifar, and Prateek Mittal. PatchCleanser: Certifiably robust defense against adversarial patches for any image classifier. In *31st USENIX Security Symposium*, pp. 2065–2082. USENIX Association, 2022.

Chong Xiang, Tong Wu, Sihui Dai, Jonathan Petit, Suman Jana, and Prateek Mittal. PatchCURE: Improving certifiable robustness, model utility, and computation efficiency of adversarial patch defenses. In *33rd USENIX Security Symposium*, pp. 3675–3692. USENIX Association, 2024.

Yuhan Xie, Yuping Yan, Yunqi Zhao, Handing Wang, and Yaochu Jin. STRONG-VLA: Decoupled robustness learning for vision-language-action models under multimodal perturbations. *arXiv preprint arXiv:2604.10055*, 2026.

Ke Xu, Yao Xiao, Zhaoheng Zheng, Kaijie Cai, and Ram Nevatia. Patchzero: Defending against adversarial patch attacks by detecting and zeroing the patch. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 4632–4641, 2023.

Idan Yankelev, Edita Grolman, Yarin Yerushalmi Levi, Amit Giloni, Omer Hofman, Toshiya Shimizu, Yuval Elovici, and Asaf Shabtai. Antistyler: Defending object detection models against adversarial patch attacks using style removal. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 27936–27945, 2026.

Brianna Zitkovich et al. RT-2: Vision-language-action models transfer web knowledge to robotic control. In *Conference on Robot Learning*, pp. 2165–2183. PMLR, 2023.

A Proofs

A.1 Proof of Lemma 1.

Fix an episode e, a policy query q, an ordered mask pair (i, j), and a tolerance $\tau \geq 0$. By Equation 6,

$$z_{e,q,i,j} = \frac{d(A_{e,q,i,j}, A_{e,q,i})}{Q_{i,j} + \epsilon}. \tag{16}$$

The calibration statistic $Q_{i,j}$ is nonnegative because it is a quantile of nonnegative disagreement values, and $\epsilon > 0$ by definition. Consequently, $Q_{i,j} + \epsilon > 0$, so multiplying an inequality by this denominator preserves its direction. We therefore have

$$\begin{aligned} z_{e,q,i,j} \leq \tau &\iff \frac{d(A_{e,q,i,j}, A_{e,q,i})}{Q_{i,j} + \epsilon} \leq \tau \\ &\iff d(A_{e,q,i,j}, A_{e,q,i}) \leq \tau(Q_{i,j} + \epsilon) \\ &\iff A_{e,q,i,j} \in \mathcal{B}_{i,j}(A_{e,q,i}; \tau). \end{aligned} \tag{17}$$

The final equivalence is precisely Definition 2, instantiated with $A = A_{e,q,i,j}$ and $B = A_{e,q,i}$. This proves the claim without requiring symmetry or the triangle inequality for d. ■

A.2 Proof of Lemma 2.

Fix an episode e and threshold $\tau \in \mathbb{R}$. Its query-index set $\mathcal{Q}_e = \{1, \dots, N_e^{\mathrm{qry}}\}$ and mask-index set $\mathcal{I} = \{1, \dots, K\}$ are finite and nonempty. Thus, all extrema in Equation 10 are attained.

First, a finite maximum is at most τ if and only if every value being maximized is at most τ. Therefore,

$$S_e \le \tau \iff \forall q \in \mathcal{Q}_e : \min_{i \in \mathcal{I}} \max_{j \in \mathcal{I}} z_{e,q,i,j} \le \tau. \tag{18}$$

Second, because the minimum over $\mathcal{I}$ is attained, for every fixed $q \in \mathcal{Q}_e$,

$$\min_{i \in \mathcal{I}} \max_{j \in \mathcal{I}} z_{e,q,i,j} \le \tau \iff \exists i \in \mathcal{I} : \max_{j \in \mathcal{I}} z_{e,q,i,j} \le \tau. \tag{19}$$

Third, for each fixed (q, i), applying the finite-maximum property again gives

$$\max_{j \in \mathcal{I}} z_{e,q,i,j} \le \tau \iff \forall j \in \mathcal{I} : z_{e,q,i,j} \le \tau. \tag{20}$$

Substituting Equation 20 into Equation 19, and then into Equation 18, yields

$$S_e \le \tau \iff \forall q \in \mathcal{Q}_e \ \exists i \in \mathcal{I} \ \forall j \in \mathcal{I} : z_{e,q,i,j} \le \tau. \tag{21}$$

By Lemma 1, the innermost condition is equivalent to $A_{e,q,i,j} \in \mathcal{B}_{i,j}(A_{e,q,i}; \tau)$. Hence, by Equation 9, the right-hand side is precisely $\mathsf{Pass}_e(\tau)$, proving both equivalences in Lemma 2. ■

A.3 Proof of Lemma 3.

First, $\mathbf{Q}$ is estimated on $\mathcal{C}_{\mathrm{scale}}$ and then fixed. Consequently, applying the same score function in Equation 10 to the n row-calibration episodes and one future clean episode preserves exchangeability. Denote the resulting scores by $S_1, \dots, S_n, S_{\mathrm{new}}$.

Second, let $S_{(1)} \le \cdots \le S_{(n)}$ be the calibration order statistics and set

$$k = \lceil (n+1)(1-\alpha) \rceil. \tag{22}$$

When $k \le n$, Algorithm 1 returns $\tau = S_{(k)}$. By exchangeability, the rank of S_{new} among all $n+1$ scores is uniform under random tie breaking. With deterministic ties, the strict failure event $S_{\mathrm{new}} > S_{(k)}$ can only become smaller, so

$$\Pr[S_{\mathrm{new}} > \tau] \le \frac{n+1-k}{n+1}. \tag{23}$$

Third, Equation 22 gives $k \ge (n+1)(1-\alpha)$ and hence $(n+1-k)/(n+1) \le \alpha$. Taking complements in Equation 23 yields

$$\Pr[S_{\mathrm{new}} \le \tau] \ge 1 - \alpha. \tag{24}$$

Finally, Lemma 2 converts $S_{\mathrm{new}} \le \tau$ into the nested clean-consistency condition stated in Lemma 3. ■

A.4 Proof of Lemma 4.

Fix an admissible support Ω and an arbitrary first mask M_i. By hypothesis, x^Ω and x agree at every pixel outside Ω. Applying the same deterministic operator M_i to both inputs cannot introduce a new difference outside Ω; therefore

$$\mathrm{supp}\big(M_i(x^\Omega) - M_i(x)\big) \subseteq \Omega. \tag{25}$$

Because $\mathcal{M}$ is $\mathcal{R}$-covering, there exists $M_{j^\star} \in \mathcal{M}$ whose mask support contains all of Ω. This second mask overwrites every pixel in the right-hand side of Equation 25 with the same fixed value in both inputs. Outside Ω, the two inputs already agree. They are therefore equal at every pixel after the second mask, which gives

$$M_{j^\star}(M_i(x^\Omega)) = M_{j^\star}(M_i(x)). \tag{26}$$

Since M_i was arbitrary, the equality holds for every first mask. ■

A.5 Proof of Theorem 1.

We assume deterministic policy inference. For stochastic policies, all predictions are conditioned on fixed seeds, so identical masked inputs produce identical action predictions. Fix any admissible support Ω. The covering definition gives an index $j^\star = j^\star(\Omega)$ that erases it. Since the returned flag is $c = \mathrm{true}$, Algorithm 2 satisfies $R_{\hat{i}} \le \tau$. By $R_i = \max_{1\le j\le K} \frac{d(A_{i,j}, A_i)}{Q_{i,j}+\epsilon}$ and Lemma 1, this is equivalent to

$$\forall j: \quad A^{\Omega}_{\hat{i},j} \in \mathcal{B}_{\hat{i},j}(A^{\Omega}_{\hat{i}}; \tau); \tag{27}$$

hence membership holds for this otherwise unknown $j^\star$. Let $A^{\Omega}_{\hat{i},j^\star} = \pi(M_{j^\star}(M_{\hat{i}}(x^{\Omega})), \ell)$. Lemma 4 gives $A^{\Omega}_{\hat{i},j^\star} = A_{\hat{i},j^\star}$ on the clean input. Since Algorithm 2 returns $\widehat{A} = A^{\Omega}_{\hat{i}}$, substitution into Equation 27 gives

$$A_{\hat{i},j^\star} \in \mathcal{B}_{\hat{i},j^\star}(\widehat{A}; \tau), \tag{28}$$

which proves Equation 14. Expanding Definition 2 gives

$$d(A_{\hat{i},j^\star}, \widehat{A}) \le \tau(Q_{\hat{i},j^\star} + \epsilon). \tag{29}$$

The first argument is patch-free by Lemma 4. Since the argument holds for every admissible Ω and $Q_{\hat{i},j^\star} \le \max_j Q_{\hat{i},j} = Q^{\max}_{\hat{i}}$, it yields the computable second inequality in Theorem 1. ∎

A.6 Proof of Theorem 2.

Let $\mathcal{T}_{\mathrm{qry}}$ be the finite set of environment times at which policy queries are issued by Algorithm 3. By its update rule,

$$C = \bigwedge_{t\in\mathcal{T}_{\mathrm{qry}}} c_t. \tag{30}$$

Assume $C = \mathrm{true}$. Equation 30 then implies

$$\forall t \in \mathcal{T}_{\mathrm{qry}}: \quad c_t = \mathrm{true}. \tag{31}$$

Fix an arbitrary $t \in \mathcal{T}_{\mathrm{qry}}$. By Equation 31, Algorithm 2 accepted some row i_t and returned its anchor chunk $\widetilde{A}_t$. Let $\Omega_t \in \mathcal{R}(P)$ be the patch support visible at this query. The hypotheses of Theorem 1 hold for the current observation and Ω_t. Therefore, an index $j^\star_t$ exists whose second mask erases Ω_t, and

$$A^{\mathrm{clean}}_{t,i_t,j^\star_t} \in \mathcal{B}_{i_t,j^\star_t}(\widetilde{A}_t; \tau). \tag{32}$$

The query time t was arbitrary, so Equation 32 holds for every $t \in \mathcal{T}_{\mathrm{qry}}$. Each application chooses $j^\star_t$ separately; hence neither the patch support nor its covering mask must remain fixed across queries. This proves the action-consistency clause of Theorem 2, including its changing-location claim.

For the conditional task-success clause, additionally assume that the task, policy, and environment satisfy Definition 3. Algorithm 3 executes the prefix of each returned $\widetilde{A}_t$, so Equation 32, which holds at every query along the actual reached trajectory, is exactly the premise of that definition. Dual-mask rollout correctness therefore implies that the complete rollout reaches the task-success set, after which the algorithm sets $S_{\mathrm{def}} = \mathrm{true}$. Since the support sequence was arbitrary subject to the threat model, the conditional implication holds for every admissible patch sequence. ∎

B Fine-tuning Design

Our fine-tuned variant, denoted OpenVLA†, is trained with dual-mask augmentation designed to improve robustness to the masking patterns used by the certification checks. For every clean demonstration $(x, \ell, A^{\mathrm{gt}})$, we sample an ordered pair (i, j) and compute $A_{i,j} = \pi(M_j(M_i(x)), \ell) \in \mathbb{R}^{H\times D}$, where H and D denote the chunk horizon and action dimension. Masking changes only the visual observation; the instruction and expert target remain unchanged.

We first regress the complete predicted chunk to its expert target:

$$\mathcal{L}_{\mathrm{task}} = \frac{1}{HD}\sum_{h=1}^{H}\sum_{d=1}^{D} \left|[A_{i,j}]_{h,d} - [A^{\mathrm{gt}}]_{h,d}\right|. \tag{33}$$

Table 3: Clean task success rates (%) under the evaluated CertVLA mask configurations on LIBERO. † denotes our dual-mask fine-tuned model.

Model	Setting	Spatial	Object	Goal	Long	Avg.
OpenVLA†	Clean	92	98	97	90	94.25
	One mask	96	100	98	95	97.25
	Two masks	95	96	92	92	93.75
OpenVLA-OFT	Clean	99	99	98	97	98.25
	One mask	92	88	98	84	90.50
	Two masks	95	88	95	80	89.50
π_0	Clean	96	98	95	79	92.00
	One mask	93	96	94	87	92.50
	Two masks	87	97	87	82	88.25
$\pi_{0.5}$	Clean	99	98	98	92	96.75
	One mask	94	99	92	91	94.00
	Two masks	97	100	98	92	96.75

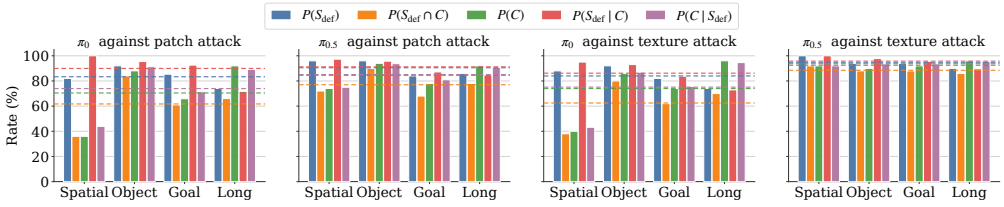


Figure 4: Certification statistics on LIBERO under patch and texture attacks. We report the defended task success rate $P(S_{\text{def}})$, the joint success-and-certification rate $P(S_{\text{def}} \cap C)$, the overall certification rate $P(C)$, the success rate among certified rollouts $P(S_{\text{def}} \mid C)$, and the certification rate among successful rollouts $P(C \mid S_{\text{def}})$, for π_0 and $\pi_{0.5}$ across the four LIBERO suites. The horizontal dashed lines denote the corresponding four-suite averages for each metric.

To preserve the direction of continuous arm motion, we additionally use

$$\mathcal{L}_{\text{dir}} = \frac{1}{H} \sum_{h=1}^{H} \left[1 - \cos\left([A_{i,j}]_{h,\text{arm}}, [A^{\text{gt}}]_{h,\text{arm}}\right)\right], \tag{34}$$

where $[A]_{h,\text{arm}}$ excludes the scalar gripper command. The final objective is

$$\mathcal{L}_{\text{ft}} = \mathcal{L}_{\text{task}} + 0.05\mathcal{L}_{\text{dir}}. \tag{35}$$

We optimize rank-32 LoRA adapters in the query while freezing the action head, proprioceptive projector, and all remaining pretrained parameters.

C Clean Task Success under Masking

Motivation. The attack-erasure guarantee identifies a dual-masked prediction that is independent of the corrupted pixels, but this reference is useful in practice only when the masked observation retains enough task information for the VLA to act successfully. We therefore evaluate clean, unattacked LIBERO rollouts under the same single- and dual-mask inputs used by CertVLA. This experiment directly measures the empirical utility of the masked reference policies appearing in Theorem 1.

Results and interpretation. Table 3 shows clean task success rate across different models under masking. The OpenVLA, which we have fine-tuned, retains 93.75% average clean success under dual masking, only 0.50 percentage points (pp) below its 94.25% clean result. Its single-mask success increases to 97.25%. We evaluate the remaining models without additional mask-specific fine-tuning to characterize the occlusion tolerance of their original checkpoints. For OpenVLA-OFT, one and two masks yield 90.50% and 89.50% average success, respectively, corresponding to

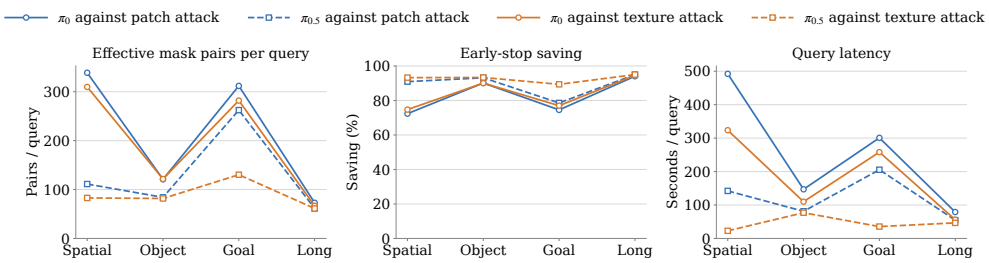


Figure 5: Efficiency of CertVLA on LIBERO under patch and texture attacks. We report the number of effective mask pairs evaluated per query (left), the fraction of mask-pair evaluations eliminated by early stopping (middle), and the resulting wall-clock latency per query (right), for π_0 and $\pi_{0.5}$ across the four LIBERO suites. Latency is measured end-to-end and includes client-server communication, batched policy inference, and hardware-dependent overhead across the GPU cluster.

decreases of 7.75 and 8.75 pp from its 98.25% clean performance. For π_0, dual masking reduces average success from 92.00% to 88.25%, while $\pi_{0.5}$ retains the same 96.75% average success with and without two masks. Overall, the complete dual-mask evaluations lose at most 8.75 pp on average. These results show that the attack-erased reference can retain task-relevant behavior on the evaluated distribution. The strong masked performance of OpenVLA† further suggests that adapting a policy to inference-time masks is a promising way to improve masked-policy utility, although this variant also uses action chunking and parallel action decoding.

D Experimental Details

D.1 Simulation Benchmark and Rollout Protocol

Each LIBERO suite contains 10 tasks. We evaluate the 10 benchmark initial states per task, yielding 100 closed-loop episodes for every model, suite, and attack setting. Unless stated otherwise, model, environment, observation, action-execution, and rollout parameters follow the official openpi (Black et al., 2024; Physical Intelligence et al., 2025) and OpenVLA-OFT (Kim et al., 2025a) protocols. Certification is evaluated online, and an episode is certified only when every policy query before termination is accepted.

D.2 Mask Construction and Evaluation Grids

All CertVLA masks overwrite a square image region with zeros in the external view. Mask positions are enumerated deterministically over the 224×224 policy input, including the final valid boundary coordinate. For patch attacks, Table 1 uses 75×75 masks for π_0 and $\pi_{0.5}$ and 50×50 masks for OpenVLA† and OpenVLA-OFT; all models use 75×75 masks for texture attacks. Calibration and defense use the same mask size for a given experimental configuration.

Calibration uses a 4×4 grid, giving $K = 16$ positions. Its stride is 50 pixels for 75×75 masks and 58 pixels for 50×50 masks, and the boundary coordinate is included in both cases. Defense uses a denser 25-pixel stride. Each defense position retrieves its entry of $\mathbf{Q}$ from the nearest calibration position. This design limits calibration cost while providing finer test-time coverage.

All evaluated configurations satisfy the deterministic covering condition

$$P \le M - s_{\text{def}} + 1, \tag{36}$$

where P, M, and s_{def} are the patch side length, mask side length, and defense stride. We enforce this condition for all experiments. Because some VLA models do not use a wrist-camera input, we define and enforce the $\mathcal{R}$-covering property (Xiang et al., 2022) only on the external-camera view. For the sensitivity study in Fig. 3, each panel reports the fixed or varied values of P, M, and the mask-set size, whose corresponding stride is determined by the evenly spaced mask grid over the 224×224 input.

D.3 Calibration Sets and Hyperparameters

Calibration uses clean, unattacked rollouts and is performed separately for each LIBERO suite. We use disjoint scale and row splits, $\mathcal{C}_{\text{scale}}$ and $\mathcal{C}_{\text{row}}$, each containing 272 calibration episodes per task from different initial seeds, or 2720 episodes per suite. Neither split overlaps the attack-evaluation episodes.

We estimate $Q_{i,j}$ as the empirical $\beta = 0.95$ quantile of clean action disagreement to reduce the influence of outliers. After fixing $\mathbf{Q}$, query scores are aggregated by the episode maximum, and τ is the finite-sample conformal quantile with $\alpha = 0.5$ to balance empirical acceptance and certificate tightness across varying episodes. Disagreement is computed over the executed actions of each predicted chunk, using normalized action bounds $[-1, 1]$ and $\epsilon = 10^{-8}$.

D.4 Shared Calibration and Policy Configuration

Models with the same embodiment and evaluation stack share calibration within each suite and mask size as an empirical approximation to reduce computational load. Specifically, π_0 and $\pi_{0.5}$ use the calibration generated by $\pi_{0.5}$, while OpenVLA† and OpenVLA-OFT generate calibration separately. This yields one fixed $(\mathbf{Q}, \tau)$ pair per model family, suite, task, and mask size. The pair is reused across patch and texture attacks whenever their mask configurations match.

For a strong and comparable clean-policy baseline, our fine-tuned OpenVLA also adopts action chunking and parallel action decoding following OpenVLA-OFT (Kim et al., 2025a). These inference improvements are used together with the dual-mask fine-tuning described in Appendix B. They improve clean task execution but do not alter the certification criterion.

D.5 Attack and Real-Robot Settings

All adversarial patches used in simulation and on the physical robot are produced using the universal transferable patch attack of UPA-RFAS (Lu et al., 2026), while adversarial object textures are generated by Tex3D (Chen et al., 2026). Patch and texture attacks use the same calibration and recovery procedure. For every evaluated attack, its visible image-plane support is bounded by a square side length P satisfying the covering condition above.

The physical evaluation uses a Piper robot with an Intel RealSense D435 external camera and a wrist-mounted Intel RealSense D405. The printed adversarial patch has the width of an A4 sheet (approximately 210 mm). CertVLA uses 75×75-pixel masks with a 25-pixel defense stride to ensure $\mathcal{R}$-covering. We evaluate the same pick-and-place task over 10 independent trials and report Clean, Attack, Defense, and Certified task success in Table 2.

D.6 Compute

Experiments are distributed across a GPU cluster and run primarily on NVIDIA A40, RTX A6000, and RTX 6000 Ada GPUs, each with 48 GB of memory.

D.7 Fine-tune Settings

We conducted dual-mask robustness fine-tuning for the LIBERO-Spatial, LIBERO-Object, LIBERO-Goal, and LIBERO-Long suites using their corresponding suite-specific models. During training, two randomly positioned 50×50 black masks were applied to the 224×224 images from both the primary and wrist cameras to simulate local visual information loss and patch- or texture-based interference. Parameter-efficient fine-tuning was performed using LoRA on the q_{proj} and v_{proj} layers, with a rank of 32 and a scaling factor of 16, while the action head and proprioceptive projector remained frozen. We used the AdamW optimizer with an initial learning rate of 2×10^{-5}, 500 linear warm-up steps, and cosine decay over 3,000 optimization steps. The effective global batch size was 48. The suite-specific policies effectively adapted to the random dual-occlusion training distribution, providing robust fine-tuned models for subsequent evaluation under patch- and texture-based perturbations.

E Analysis of Certified Rollouts

Fig. 4 separates task utility from certificate coverage. Let S_{def} denote defended task success and C denote that every query in a rollout is certified. Defense is $P(S_{\mathrm{def}})$, while the Certified metric in the main tables is the joint rate $P(S_{\mathrm{def}} \cap C)$. The raw rate $P(C)$ measures how often complete rollouts admit the formal bound, irrespective of task outcome. $P(S_{\mathrm{def}} \mid C)$ measures success among certified rollouts, whereas $P(C \mid S_{\mathrm{def}}) = P(S_{\mathrm{def}} \cap C)/P(S_{\mathrm{def}})$ is the fraction of successful defended rollouts backed by an end-to-end certificate. An uncertified success is therefore a certification abstention: the rollout may still succeed empirically, but CertVLA does not attach its formal guarantee to that execution.

Across the four LIBERO suites, $P(C)$ averages 70.5% and 84.5% for π_0 and $\pi_{0.5}$ under patch attacks and 74.0% and 92.5% under texture attacks. Certified rollouts also have high task utility: $P(S_{\mathrm{def}} \mid C)$ averages 89.9%, 91.3%, 86.2%, and 95.8% for the four model-attack settings. The respective gaps between $P(C)$ and $P(S_{\mathrm{def}} \cap C)$ are 8.8, 8.0, 11.5, and 4.0 pp because certification verifies clean-masked action consistency rather than terminal success, and masking can make even an attack-free policy fail. Conversely, $P(C \mid S_{\mathrm{def}})$ averages 74.0% and 84.3% under patch attacks and 75.1% and 93.7% under texture attacks. Thus, the certificate covers a large fraction of successful executions while remaining conceptually distinct from task success.

F Efficiency Analysis

Fig. 5 analyzes the computational overhead of CertVLA. Across all model-attack settings, early stopping eliminates approximately 72%–95% of the candidate mask-pair evaluations, substantially reducing the number of pairs that need to be explicitly processed. Early stopping does not change whether a row satisfies the complete certification test, although the uncertified fallback action can differ when no row passes. The resulting computation varies across tasks: Spatial and Goal generally require more effective mask pairs, whereas Long requires the fewest. For example, π_0 requires roughly 300–340 effective pairs on Spatial and Goal in several settings, while the number drops below 100 on Long.

The efficiency gain is particularly pronounced for $\pi_{0.5}$. Compared with π_0, it consistently requires fewer effective mask pairs and achieves lower query latency, especially on Spatial and Goal. Under texture attacks, for instance, the effective pair count for $\pi_{0.5}$ is reduced to roughly 80–130 pairs across the four suites, compared with up to about 300 pairs for π_0. Despite these differences, the early-stop saving remains high across both models and both attack types. The reported wall-clock time is an end-to-end system measurement that includes client-server communication, batched policy inference, and other execution overhead. Because experiments are distributed across A40, RTX A6000, and RTX 6000 Ada GPUs, these values also include hardware-dependent variation and should not be interpreted as hardware-normalized model inference latency.

G Overall Interpretation

What does the certificate mean physically? The key intuition is that masking can create an input whose prediction is independent of the attack content. For every patch satisfying our bounded-support threat model, $\mathcal{R}$-covering guarantees that at least one tested mask $M_{j^\star}$ covers the entire patch. Overwriting those pixels makes the masked attacked observation exactly equal to the corresponding masked clean observation. Its action prediction is therefore an *attack-erased reference*: it contains black occlusion, but it contains no information introduced by the attacker.

The policy naturally changes its action when clean observations are masked, so exact equality with the original clean action would be unnecessarily strict. We instead measure this benign variation on clean data and calibrate the clean-consistency region $d(A_{i,j}, A_i) \le \tau(Q_{i,j} + \epsilon)$. This region can be viewed informally as a "safe range," meaning a range of action variation that is consistent with clean masked behavior. At test time, an anchor action is certified only if it is inside the corresponding region for *every* second mask. The universal check necessarily includes $M_{j^\star}$, thereby linking the accepted anchor to an attack-erased reference.

Scope of the guarantee. For a certified query, any adaptive attacker satisfying the support bound cannot move the returned action outside the calibrated region around at least one attack-erased prediction. Requiring this property at every query extends the statement to the closed-loop action sequence. The guarantee directly concerns attack-erased action consistency. Converting it into unconditional task correctness additionally requires dual-mask rollout correctness, because even an attack-free masked policy can occasionally choose a task-failing action. These measurements quantify the empirical task utility of certified rollouts, but they do not establish the universal dual-mask rollout-correctness condition assumed by the conditional task-success clause of Theorem 2.