Can Pixels Alone Reveal Image Origin? Minimax Limits and Learnable Interfaces for Passive Provenance

Kai Yao
School of Informatics
University of Edinburgh
kai.yao@ed.ac.uk

Abstract

Passive image provenance asks whether pixels alone can reveal where an image came from: a human, an aggregate AI class, or a particular generator. This becomes a robustness problem once a source image can be edited before the verifier sees it. We study the problem as source–target verification under adversarial distribution shift. Our first result gives the exact best-case limit for any image-only verifier: the largest robust target-acceptance gap equals the minimum total-variation distance between the target distribution and the set of attacked source distributions. This quantity depends on the source, target, and edit class, not on the verifier architecture. Our second result explains why deployed public verifiers can fail before this statistical limit is reached. If the verifier can be emulated on the attack region to error ε, then a surrogate black-box attack reaches target acceptance within 2ε plus optimization error of the white-box optimum; score-revealing logistic and softmax heads over public features are identifiable, and approximate score access gives stable recovery bounds. A finite-state experiment checks the minimax identity where both sides are computable. On same-prompt real/diffusion benchmarks, the evaluated public CLIP verifiers fail under targeted pixel attacks, while a ResNet-18 victim exhibits partial fake-to-real transfer. Binary feedback with abstention reduces measured attack success, but positive empirical gap upper bounds do not establish robustness. These results motivate separate evaluation of the source–target statistical ceiling and the information released by a deployed verifier.

1 Introduction

A passive image-provenance system receives an image and must infer its origin from pixels alone, continuing a forensic line that uses sensor and camera-model traces [1, 2] and learned forensic embeddings [3]. It may ask whether the image is real or AI-generated [4–7], or it may ask which generator produced it [8, 10, 11]. This looks like a classification problem on clean data. Under adaptive editing, detector-evasion and adversarial-example results show that image edits can alter a classifier's decision [12–20]. We focus on targeted forgery: an edited source image is intended to be accepted as a chosen target.

We use the following source–target view throughout the paper. A source image is drawn from a source distribution P_s. The attacker may edit it within an admissible set $\Gamma(x)$, where x is the original image. The verifier outputs a target score $f_t(x) \in [0, 1]$, interpreted as acceptance of target label t. The central question is: how much larger can the target score be on true target images than on any edited source images?

There are two distinct reasons this separation can fail. The first reason is statistical. If the edited source distribution can get close to the target distribution, then no image-only verifier can reliably

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

separate them. The second reason is interface-based. Even when image-level signal remains, a particular public verifier may reveal enough information for an attacker to learn a surrogate and optimize against it. These two failures look similar in experiments, but they require different fixes: better training cannot beat the statistical ceiling, while interface design can reduce learnability of the deployed verifier.

Main take-away. The paper separates these two mechanisms. Barrier I is an exact minimax ceiling: the best possible passive verifier is limited by the total-variation distance from the target distribution to the attacked source family. Barrier II is a deployed-verifier failure mode: a learnable public interface can make black-box attacks nearly white-box. In the evaluated CLIP interface comparison, binary feedback with abstention reduces attack success; its positive empirical gap upper bounds leave the true robust gap unresolved.

Contributions.

1. **Exact minimax limit for target verification.** We define the attackability radius
$$\Delta_{s,t} := \inf_{Q \in \mathcal{Q}_s} \mathrm{TV}(P_t, Q),$$
where $\mathcal{Q}_s$ is the set of distributions reachable by editing source samples and TV is total variation. We prove that
$$\sup_{f_t : \mathcal{X} \to [0,1]} \left(\mathbb{E}_{P_t} f_t - \sup_{Q \in \mathcal{Q}_s} \mathbb{E}_Q f_t \right) = \Delta_{s,t}.$$
Thus the optimal passive verifier is governed by a distributional projection of the target onto the attacked source family.
2. **Verifier-learnability barrier for public interfaces.** We prove that if a deployed verifier is ε-emulable on the attack region and the attacker optimizes the emulator with expected error $\bar{\alpha}$, then black-box targeted attacks achieve target acceptance within $2\varepsilon + \bar{\alpha}$ of the white-box optimum.
3. **Exact and stable recovery of score-revealing heads.** We specialize established score-based recovery ideas to logistic and softmax heads over public features. Full score access reduces the attack to system identification; bounded log-odds error gives a stable emulator when the probe features are well conditioned.
4. **Query-cost separation for private low-bandwidth interfaces.** We prove a conditional query bound in a stylized hidden-state model, showing how private low-bandwidth feedback can raise attack cost. This does not change the image-only statistical ceiling.
5. **Empirical evidence on same-prompt real/diffusion benchmarks.** We evaluate public, limited-public, and private interfaces. The evaluated public CLIP verifiers retain negative empirical gap upper bounds after stronger clean or adversarial retraining. We add a finite-state check of the minimax identity, partial fake-to-real transfer to a ResNet-18 victim, and paired image-quality measurements.

Source code: `https://github.com/kaikaiyao/pixels-alone-provenance`.

2 Related Work

Adversarial robustness. Adversarial robustness studies classifiers under worst-case perturbations, robust optimization, transfer, black-box attacks, and gradient-based attacks [12–16]. We use the standard variational characterization of total variation and Sion's minimax theorem to characterize the largest robust target-verification gap achievable by any image-only verifier against a family of edited source distributions. Model extraction [58], knowledge distillation [59], and surrogate attacks [13] are established tools; our emulator theorem quantifies the effect of score emulation on deployed target acceptance within this source–target analysis.

Passive provenance, detection, and attribution. Passive provenance follows the forensic idea that origin traces may remain in the released artifact. Prior work used sensor and camera-model traces [1, 2] and learned forensic embeddings for manipulation detection [3]; other work studies face-manipulation detection [21–24], synthetic-image detection and forensic artifacts [4–7, 25–31], and

generator attribution/open-set attribution [8, 10, 11, 33–38]. Prior work shows useful clean accuracy and many empirical failures under attack [17–20, 39]. Our contribution is an exact source–target minimax characterization and a separate emulator-to-attack bound for a deployed verifier. Neves et al. [52] remove GAN fingerprints to evade fixed face-manipulation detectors; Mavali et al. [53] study transfer- and query-based attacks on selected AI-image detectors; Wang et al. [54] attribute images through accessible candidate generators and evaluate filter and blur edits. We build on this fixed-system literature to distinguish a source–target limit over all image-only verifiers from the learnability of one deployed interface; the surrogate attack itself is not claimed as new.

Watermarking and active provenance. Watermarking modifies the generator, training process, or generated artifact to support later detection or attribution [9, 32, 40–44]. This is complementary to our setting. We study passive verification, where no trusted generator-side mark or metadata is assumed. For language models, Zhang et al. [45] prove strong-watermark impossibility under quality-oracle and efficiently mixing perturbation-oracle assumptions; for images, Fairoze et al. [46] study the difficulty of jointly achieving robustness, unforgeability, and public detectability. Our results isolate another mechanism: even if image-level signal exists, the released verifier interface may itself become the attack surface.

Private auditors. Keyed passive attribution methods such as SPRINT keep the verifier's reconstruction task private to limit the information available to an adversary [47]. Our private-interface theorem is not a literal model of any one auditor. It isolates a query-complexity mechanism: hidden verifier state and low-bandwidth feedback can force search instead of direct score optimization.

3 Target Verification under Adversarial Distribution Shift

We now define the notation used in the rest of the paper. The image space $\mathcal{X}$ is finite, which matches quantized digital images and keeps the main proof elementary. Appendix F gives an extension to general measurable image spaces under the stated total-variation compactness assumption. Labels are denoted by $y \in \mathcal{Y}$; a label may be HUMAN, an aggregate AI class, or a particular generator. The distribution of images with label y is P_y.

For a source–target task, s is the source label and t is the target label. A target verifier is a score function

$$f_t : \mathcal{X} \to [0, 1],$$

where $f_t(x)$ is the probability or confidence with which image x is accepted as target t. If $d : \mathcal{X} \to [0, 1]$ is a binary AI detector, then $f_{\mathsf{A}} = d$ and $f_{\mathsf{H}} = 1 - d$. If $g : \mathcal{X} \to \mathcal{P}(\{1, \dots, m\})$ is a multiclass attribution model, then $f_t(x) = g_t(x)$.

The edit correspondence $\Gamma : \mathcal{X} \to 2^{\mathcal{X}} \setminus \{\emptyset\}$ specifies which edited outputs are allowed from each source image. A stochastic editor is a Markov kernel $A(\cdot \mid x)$ supported on $\Gamma(x)$. Applying this editor to $X \sim P_s$ gives an attacked source distribution $A_{\#}P_s$. We collect all such distributions as

$$\mathcal{Q}_s := \{A_{\#}P_s : \operatorname{supp}(A(\cdot \mid x)) \subseteq \Gamma(x)\ \forall x \in \mathcal{X}\}.$$

The target label changes the objective, but not the reachable family $\mathcal{Q}_s$. The set $\mathcal{Q}_s$ is nonempty, convex, and compact; the short proof is in Appendix A.

For a fixed pair (s, t), define clean target acceptance and worst-case targeted acceptance by

$$\mathrm{Acc}_t(f_t) := \mathbb{E}_{P_t}[f_t], \qquad W_{s\to t}(f_t; \Gamma) := \sup_{Q \in \mathcal{Q}_s} \mathbb{E}_Q[f_t].$$

The robust target-verification gap is

$$\mathrm{Gap}_{s,t}(f_t) := \mathrm{Acc}_t(f_t) - W_{s\to t}(f_t; \Gamma).$$

A positive gap means that true target images receive higher average target score than any edited source distribution. A negative gap means an edited source can receive even higher target score than clean target samples.

For an aggregate AI label, we write $P_{\mathsf{A}} = \sum_j \pi_j P_j$, where P_j is the distribution of generator j and the weights π_j describe the deployment mixture. This mixture notation is used only when discussing coarse AI-vs-human verification.

4 Barrier I: A Minimax Statistical Ceiling

Barrier I asks the most favorable question for the defender: if we could choose any image-only verifier, what robust gap could we guarantee? Define the attackability radius

$$\Delta_{s,t} := \inf_{Q \in \mathcal{Q}_s} \mathrm{TV}(P_t, Q), \qquad \mathrm{TV}(P, Q) := \frac{1}{2} \sum_{x \in \mathcal{X}} |P(x) - Q(x)|.$$

This is the distance from the target distribution to the closest attacked source distribution.

Theorem 1 (Attackability-radius law)**.** *For every source–target pair* (s, t)*,*

$$\sup_{f_t : \mathcal{X} \to [0,1]} \mathrm{Gap}_{s,t}(f_t) = \Delta_{s,t}.$$

Equivalently,

$$\sup_{f_t : \mathcal{X} \to [0,1]} \inf_{Q \in \mathcal{Q}_s} (\mathbb{E}_{P_t} f_t - \mathbb{E}_Q f_t) = \inf_{Q \in \mathcal{Q}_s} \mathrm{TV}(P_t, Q).$$

Why this theorem matters. The theorem says that the best possible passive verifier is determined by the geometry of distributions, not by a model class or a training algorithm. If edited source distributions can approach the target distribution, no passive image-only method can keep a large robust gap.

Proof sketch. For a fixed attacked distribution Q, the best $[0, 1]$-valued test separates P_t from Q with value exactly $\mathrm{TV}(P_t, Q)$. This is the standard variational form of total variation. The nontrivial step is that the attacker is not fixed: it may choose any $Q \in \mathcal{Q}_s$. Since $\mathcal{Q}_s$ and $[0, 1]^{\mathcal{X}}$ are compact convex sets and the payoff is affine in both arguments, Sion's minimax theorem [48] lets us exchange the verifier and attacker optimizations. The value becomes the closest total-variation distance from P_t to $\mathcal{Q}_s$. The full proof, including the compactness and total-variation lemmas, is in Appendix A.

Corollary 2 (Universal passive ceiling)**.** *For every passive image-only verifier* f_t*,*

$$\mathrm{Gap}_{s,t}(f_t) \leq \Delta_{s,t}.$$

Thus if $\Delta_{s,t} = 0$*, no passive verifier can maintain positive robust target separation under the edit class* Γ*.*

Proposition 3 (Coarsening can lower the target ceiling)**.** *If an aggregate target is* $P_{\mathsf{A}} = \sum_j \pi_j P_j$*, then for any source label* s*,*

$$\Delta_{s,\mathsf{A}} \leq \sum_j \pi_j \Delta_{s,j}.$$

Interpretation. A coarse label such as AI is a mixture of generator-specific targets, so it can be easier to hit than a particular generator target. This is why binary detection and model attribution need not have the same robustness profile. The proof of Proposition 3, and finite examples showing that human-evasion and model-impersonation radii have no universal ordering, are in Appendix A. Appendix E explains how this statistical ceiling relates to the empirical gap upper bounds reported later: the image experiments diagnose failure on the evaluated samples without directly estimating $\Delta_{s,t}$.

5 Barrier II: Learnable Interfaces Collapse to Surrogate Attacks

Barrier II studies a different object: a fixed deployed verifier. Even when the statistical ceiling is not tight, a public interface may expose enough information to learn a useful surrogate. Let the attack region be

$$\mathcal{U} := \bigcup_{x \in \mathcal{X}} \Gamma(x).$$

A surrogate $\widehat{f_t} : \mathcal{U} \to [0, 1]$ is an ε-emulator of f_t on $\mathcal{U}$ if

$$\|\widehat{f_t} - f_t\|_{\infty,\mathcal{U}} := \sup_{u \in \mathcal{U}} |\widehat{f_t}(u) - f_t(u)| \leq \varepsilon.$$

For an attack map $\widehat{a}(x) \in \Gamma(x)$, define its surrogate optimization loss

$$\alpha(x) := \sup_{z \in \Gamma(x)} \widehat{f}_t(z) - \widehat{f}_t(\widehat{a}(x)), \qquad \bar{\alpha} := \mathbb{E}_{X \sim P_s}[\alpha(X)].$$

Here $\bar{\alpha}$ is the average loss of the attack optimizer on the surrogate objective.

Theorem 4 (Surrogate-attack theorem)**.** *If $\widehat{f}_t$ is an ε-emulator of f_t on $\mathcal{U}$, then any attack map $\widehat{a}$ satisfies*

$$\mathbb{E}_{X \sim P_s}[f_t(\widehat{a}(X))] \geq W_{s \to t}(f_t; \Gamma) - 2\varepsilon - \bar{\alpha}.$$

Intuition and proof sketch. There are only three losses. The surrogate may score the true white-box maximizer incorrectly by at most ε. The optimizer may lose $\alpha(x)$ on the surrogate. Finally, when the chosen point is evaluated by the deployed verifier, the surrogate may again differ by at most ε. Averaging over source samples gives the bound. The full proof is in Appendix B.

Corollary 5 (Learned emulators suffice)**.** *Suppose a query procedure returns a random surrogate $\widehat{f}_t$ such that*

$$\mathbb{P}[\|\widehat{f}_t - f_t\|_{\infty, \mathcal{U}} \leq \varepsilon] \geq 1 - \delta.$$

Then any attack map optimized on the returned surrogate with expected loss $\bar{\alpha}$ satisfies

$$\mathbb{P}\left[\mathbb{E}_{X \sim P_s}[f_t(\widehat{a}(X))] \geq W_{s \to t}(f_t; \Gamma) - 2\varepsilon - \bar{\alpha}\right] \geq 1 - \delta.$$

This corollary is the main deployment message of Barrier II. Exact parameter recovery is not required. Any learning procedure that gives a good emulator on the attack region gives a near-white-box targeted attack.

Proposition 6 (Score-revealing public heads are identifiable)**.** *Let*

$$g_H(j \mid x) = \frac{\exp(\langle h_j, \phi(x) \rangle)}{\sum_{\ell=1}^{m} \exp(\langle h_\ell, \phi(x) \rangle)}$$

be a softmax attribution head over a public feature map $\phi : \mathcal{X} \to \mathbb{R}^p$, with gauge $h_m = 0$. If the interface returns the full probability vector and the attacker can query p points whose feature matrix is invertible, then the head parameters $H = (h_1, \ldots, h_m)$ are exactly recoverable from p queries. The binary logistic case is the $m = 2$ specialization. Consequently, surrogate attacks on the recovered head match white-box attacks up to optimization error.

Why this proposition is useful. A score-revealing softmax head exposes log-odds. With public features, those log-odds are linear equations in the unknown head weights. Once the probe features span $\mathbb{R}^p$, recovering the verifier is just solving linear systems. The detailed derivation is in Appendix B.

Proposition 7 (Stable recovery under logit error)**.** *In the setting of Proposition 6, suppose the recovered log-odds at the p probe points have error at most η per coordinate, and let Φ be the probe feature matrix. Then each recovered class vector satisfies*

$$\|\widehat{h}_j - h_j\|_2 \leq \|\Phi^{-1}\|_2 \sqrt{p}\, \eta.$$

If $\|\phi(u)\|_2 \leq R$ on the attack region, the induced logit error per class on the attack region is at most

$$B := R\|\Phi^{-1}\|_2 \sqrt{p}\, \eta.$$

Thus each recovered target score $\widehat{f}_t$ is an ε-emulator of f_t, with $\varepsilon \leq B/2$ for softmax, or $\varepsilon \leq B/4$ for binary logistic in the reference-class gauge.

Interpretation. The exact-recovery assumption is not a knife-edge. With bounded log-odds error, recovery degrades with the conditioning of the probe matrix and the feature norm on the attack region. Deriving such a bound from noisy or quantized probabilities additionally requires the probabilities entering the ratios to stay bounded away from zero. Thus finite-precision score access can still yield the emulator required by Theorem 4. A proof is in Appendix B.

The recovery propositions require score-revealing interfaces and public features. They do not claim that every public verifier is exactly recoverable, nor that top-1 labels alone always reveal enough information. They are concrete instantiations of the broader learnability barrier.

6 Private Interfaces Raise Query Cost, Not the Statistical Ceiling

Barrier I applies to private and public image-only verifiers. Privacy cannot create provenance information that is absent from the image. What it can change is how hard it is for the attacker to learn and optimize the verifier. We isolate this effect with a simple hidden-state search bound.

Theorem 8 (Low-bandwidth hidden-state search)**.** *Let Θ be a finite hidden verifier-state space, with $\theta \sim \mathrm{Unif}(\Theta)$. Suppose a q-query interface returns one of at most M symbols per query. After observing the transcript, the attacker outputs an action a. If every fixed action succeeds on at most a ρ fraction of hidden states,*

$$\sup_{a} \Pr_{\theta \sim \mathrm{Unif}(\Theta)}[\mathrm{success}(a, \theta)] \le \rho,$$

then every adaptive q-query attacker has success probability at most

$$M^q \rho.$$

Intuition. After q queries, the attacker can see at most M^q transcripts. Each transcript leads to one final action. If any fixed action only works on a ρ fraction of hidden states, then the union over all transcripts covers at most an $M^q\rho$ fraction. The conclusion depends on this fixed-action success bound as well as the interface alphabet; low bandwidth alone does not imply security. The full counting proof is in Appendix C.

Corollary 9 (Hidden-support threshold verifier)**.** *In the hidden-support model with secret $S \subseteq [d]$, $|S| = k$, candidate budget b, and threshold τ, a binary interface satisfies*

$$\Pr[\mathrm{success}] \le 2^q \frac{\sum_{i=\tau}^{\min\{k,b\}} \binom{b}{i}\binom{d-b}{k-i}}{\binom{d}{k}}.$$

In particular,

$$\Pr[\mathrm{success}] \le 2^q \binom{b}{\tau} \left(\frac{k}{d}\right)^{\tau}.$$

The hidden-support model is deliberately simple. It should be read as a clean example of the hidden-state theorem, not as a literal model of a deployed provenance service. With abstention, the interface alphabet may grow, but the accepting region can shrink; this is why abstention can reduce attack success even though it adds another possible output symbol. Details and the sparse-support query lower bound are in Appendix C.

7 Empirical Evaluation

The image experiments keep the source–target task fixed and vary what the verifier reveals: full score vectors, single target scores, quantized scores, top-1 labels, binary decisions, and binary decisions with abstention. They measure attacks on fixed verifiers, while a separate finite-state experiment checks Barrier I where its minimax value is computable. Appendix D reports the original per-direction results and controls; Appendix H gives the attack protocol and additional evaluations. The interfaces are local API-style abstractions; no live commercial service is evaluated.

Benchmarks and verifiers. We use two same-prompt benchmarks. The controlled three-class benchmark uses 1,200 MS COCO prompts [49] to create aligned triplets of real, Stable Diffusion 1.5 [50], and SDXL images, with a prompt-level split of 600/300/300 for train/validation/test and 900 held-out test images. The expanded benchmark keeps the same prompt protocol and adds Stable Diffusion 2.1 and PixArt. Verifiers include frozen CLIP [51] representations with learned linear or MLP heads, and off-the-shelf CLIP-based detector baselines. Attack success rate (ASR) is the fraction of attacked source images accepted as the target under the evaluated interface. The additional ResNet-18 and image-quality evaluations also report induced success among initially correct sources, excluding pre-existing target predictions.

Attack protocol. The attacker recovers or fits an emulator from interface replies, runs targeted projected gradient descent (PGD) through the public feature extractor and emulator, and evaluates the final images on the deployed verifier. Pixel attacks use ℓ_∞ budgets $2/255$, $4/255$, or $8/255$, 40 steps, step size one quarter of the budget, and five restarts. Appendix H.1 specifies the interface-dependent losses and transcript counts.

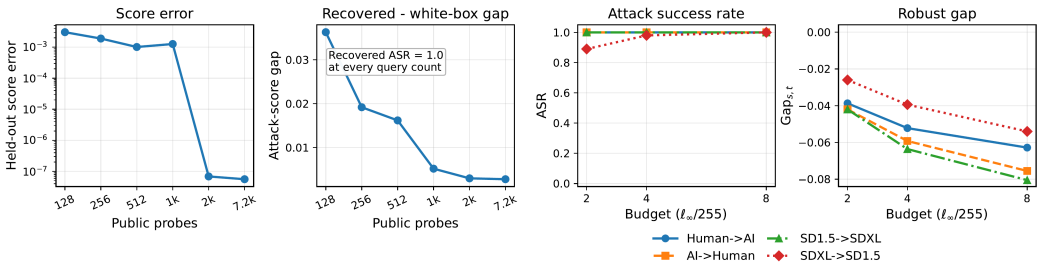


Figure 1: Controlled public CLIP-softmax verifier on the same-prompt three-class benchmark. Score-revealing queries recover the deployed score function; attacks on the recovered verifier approach white-box performance; all four source–target directions collapse; and every estimated gap upper bound is negative on the evaluated samples.

Empirical upper bound on the robust gap. For an evaluated attack routine $\widetilde{a}$, we report

$$\widetilde{\mathrm{Gap}}_{s,t} := \mathrm{Acc}_t(f_t) - \mathbb{E}_{X\sim P_s}[f_t(\widetilde{a}(X))].$$

Because $W_{s\to t}(f_t;\Gamma) \geq \mathbb{E}[f_t(\widetilde{a}(X))]$, we have

$$\mathrm{Gap}_{s,t}(f_t) \leq \widetilde{\mathrm{Gap}}_{s,t}.$$

This inequality holds for population expectations: a negative attack-derived upper bound certifies a negative robust gap for that fixed verifier, whereas a positive bound leaves its sign unresolved. The image tables estimate these expectations by held-out sample averages and retain the notation $\widetilde{\mathrm{Gap}}$; their signs describe the evaluated samples, without a population confidence guarantee. They do not estimate the minimax radius $\Delta_{s,t}$ (Appendix E).

Controlled check of Barrier I. On $\mathcal{X} = \{0,1\}^6$, five source–target pairs and Hamming radii $0, 1, 2, 3$ give 20 settings. Independent attack-side transport and verifier-side minimax linear programs agree to a maximum absolute discrepancy of 4.44×10^{-16}. Across 20 repeats per setting, plug-in mean absolute error falls from 0.02094 at $n = 250$ to 0.00451 at $n = 5{,}000$ (Table 10). This checks the computable finite-state identity; the radius for realistic images remains unestimated.

7.1 Public score access collapses the controlled verifier

The controlled three-class CLIP-softmax verifier is public, score revealing, and simple enough to instantiate Proposition 6. It is not a vacuous model: on the held-out test set it reaches 0.5167 accuracy and 0.4802 macro-F1 on the fine-grained three-way task. The induced fake-vs-real detector reaches fake AUC 0.8872, TPR@5%FPR 0.6183, and EER 0.2192.

At 7,200 probes, the observed score error is 5.60×10^{-8}, and attacks on the recovered head differ from direct white-box attacks by 0.0028 in target score. By 2,048 queries, score error is already 6.88×10^{-8} and the recovered-vs-white-box gap is 0.0030. Attacks optimized on the recovered verifier achieve ASR 1.0. These measurements instantiate Proposition 6: public score vectors turn black-box attack into head identification followed by white-box optimization. Figure 1 reads from left to right as score recovery, difference from direct white-box optimization, directional ASR, and empirical gap upper bounds. At $8/255$, Human→AI has clean-target and attacked-source scores 0.6786 and 0.7414 (reported gap -0.0628); AI→Human has 0.3338 and 0.4094 (reported gap -0.0755). Reversing direction changes both distributions and target scores; the two gaps need not coincide. Table 4 gives all directions.

Three source–target directions reach ASR 1.0 at $2/255$: Human→AI false-flagging, AI→Human evasion, and SD1.5→SDXL impersonation. SDXL→SD1.5 rises from ASR 0.89 at $2/255$ to 1.0 by $8/255$. Every empirical upper bound on the robust gap is negative, so edited source images exceed the target label's own clean acceptance, not merely its decision boundary.

Table 1: Representative $8/255$ interface slice on the expanded five-class benchmark. Entries are ASR; a dash denotes an unevaluated combination. Plain binary feedback remains attackable for both learned heads. Binary feedback with abstention uses the configured per-split cap $7{,}200$ and reduces ASR across these three systems.

Verifier	Full vector	Single score	Top-1 label	Binary verdict	Binary + abstain
Linear softmax CLIP head	1.00	0.91	0.75	1.00	0.00
Two-layer MLP CLIP head	1.00	0.93	1.00	1.00	0.00
Off-the-shelf CLIP detector	1.00	1.00	1.00	–	0.09

Table 2: Released and retrained models on the original three-class benchmark. Clean columns report test accuracy and fake AUC; attack columns summarize the minimum ASR, maximum (best) gap, or minimum (worst) gap across the four evaluated source–target directions at the indicated budget.

Model	**Acc.**	**AUC**	Min ASR $2/255$	Min ASR $8/255$	Best gap $2/255$	Worst gap $8/255$
Released	0.517	0.887	0.890	1.000	-0.026	-0.081
Clean retrain	0.869	0.982	0.967	1.000	-0.169	-0.364
Adv. retrain	0.821	0.947	0.900	1.000	-0.024	-0.185

7.2 Interface limits matter more than clean strength

The expanded five-class benchmark tests whether changing the released interface changes attackability. Table 1 shows the main pattern. Full vectors and single scores remain highly attackable. Top-1 labels reduce ASR for the linear head but not for the nonlinear head or the off-the-shelf CLIP detector. Plain binary feedback gives ASR 1.0 for both learned heads, with estimated gap upper bounds -0.0260 and -0.0498. Binary feedback with abstention reduces ASR and yields positive estimated upper bounds in this CLIP comparison; these positive values do not certify robustness. Appendix D reports clean performance, transcript-cap sweeps, and a score-guided random-search control.

For the shared-representation CLIP heads, the validation-set 95th-percentile emulator diagnostic ϵ_{95} rises from about 0.005 under full vectors to 0.31–0.33 under a single score, about 0.70 under plain binary feedback, and 1.0 under top-1 labels or abstention. This diagnostic is not the uniform attack-region error ε in Theorem 4; the sweep also does not estimate $\bar{\alpha}$, so it does not numerically validate the $2\varepsilon + \bar{\alpha}$ bound. High validation error alone does not imply safety: top-1 attacks still succeed against the MLP and off-the-shelf detector.

7.3 Stronger retraining still does not restore robustness

A natural objection is that public-interface collapse is caused by weak clean training. Table 2 tests this directly on the original three-class benchmark. The stronger clean retrain improves accuracy from 0.517 to 0.869 and fake AUC from 0.887 to 0.982, but its empirical gap upper bounds remain negative. Adversarial retraining reduces the severity of some attacks, but the estimated gap upper bounds remain negative at $8/255$. Supporting per-direction results are in Table 9.

7.4 Transfer beyond CLIP and paired image quality

An RGB ResNet-18 [55] victim reaches test fake AUC 0.9862. At $8/255$, attacks through a CLIP-feature emulator induce fake-to-real success on $45/117$ initially correct sources (38.46%), but real-to-fake success on $0/122$. Both estimated gap upper bounds remain positive (Table 11). This supports partial directional transfer across representations for one victim and setting, rather than general collapse beyond CLIP.

In a paired evaluation of four attack directions, each budget has 64 attack instances, of which 50 start from correct predictions. At $2/255$, $45/50$ are induced successes, with pooled success-only medians of 0.9784 structural similarity (SSIM) [57], 0.0063 learned perceptual image patch similarity (LPIPS-Alex) [56], and 43.82 dB peak signal-to-noise ratio (PSNR). At $8/255$, induced success

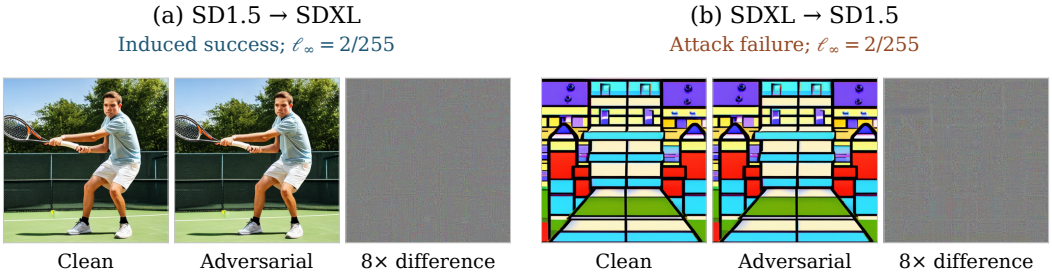


Figure 2: Paired examples at an ℓ_∞ budget of $2/255$. Each triplet shows the clean image, the attacked image, and the signed perturbation amplified eightfold around mid-gray. The SD1.5→SDXL example induces target acceptance, while the SDXL→SD1.5 example fails. These selected examples illustrate both outcomes.

reaches $50/50$, while per-direction median SSIM falls to 0.8011–0.8534 and LPIPS-Alex rises to 0.0770–0.1365 (Table 12). The larger budget trades paired image quality for five additional successes. Figure 2 shows a success and a failure; Appendix H.4 includes paired examples at both budgets. These examples and metrics do not constitute a human perceptual study.

8 Discussion and Limitations

The paper separates two failure modes that are easy to confuse. The attackability-radius law is an image-only statistical ceiling: it applies even to the best possible passive verifier and depends only on P_t, P_s, and Γ. The surrogate-attack theorem is an interface statement: it explains how one deployed public verifier can fail before that ceiling is reached.

The experiments support the second mechanism in the evaluated settings. Public CLIP-based interfaces are learnable enough that targeted attacks reach or exceed the target label's own clean acceptance. Stronger clean retraining and adversarial retraining improve clean metrics but do not restore positive empirical upper bounds on the robust gap.

What Barrier I does and does not say. Barrier I is an optimistic benchmark for passive verification. It asks what would be possible if the defender could choose the best possible image-only verifier, ignoring architecture, computation, and training data. A small $\Delta_{s,t}$ means the edited source family can get close to the target distribution, so there is little robust separation available to any passive method. A large $\Delta_{s,t}$ does not mean a deployed model is safe. It only means that the image distributions leave room for a good verifier in principle.

What Barrier II adds. Barrier II explains how a real verifier can lose that room. The theorem does not require the attacker to match the target distribution. It only requires the attacker to learn enough of the deployed score on the attack region. This is why the empirical upper bound $\widetilde{\mathrm{Gap}}$ is useful: when it is negative, the attacked source samples already receive higher target score than clean target samples under the deployed verifier. That is a failure of the deployed verifier, even if the unknown statistical ceiling $\Delta_{s,t}$ is not directly estimated.

What the private-interface results mean. The private-interface theorem is conditional on hidden verifier state and the fixed-action success bound ρ. It shows how limited transcripts can constrain search under those assumptions. In the CLIP experiments, abstention changes the accepting region and reduces measured attack success; its positive estimated gap upper bounds leave robustness unresolved. Such interface restrictions cannot overcome a small attackability radius.

How to use the results. For a new passive provenance system, clean accuracy should be treated as only a first diagnostic. A more relevant evaluation should report source–target attack directions, the interface exposed to the attacker, attack success rate, and the empirical upper bound $\widetilde{\mathrm{Gap}}$ on the robust gap. If public scores are released, the system should also be tested for surrogate learnability. If a private interface is used, the evaluation should test whether its accepting region can still be

discovered through queries. The appendices give the full proofs and the extra empirical tables needed to audit these claims.

What should not be inferred. Our results do not say that passive provenance is useless. They say that a passive verifier has two separate obligations. First, the source and target distributions must remain separated after the allowed edits; this is the Barrier I question. Second, the deployed interface must not hand the attacker a useful optimization signal; this is the Barrier II question. A system can have strong clean accuracy and still fail the second obligation. Conversely, a private interface can make attacks harder while still leaving the first obligation unresolved.

Several limitations remain. First, the finite-state calculation does not estimate $\Delta_{s,t}$ for realistic image distributions; Appendix E describes bounds whose tightness remains unresolved. Second, the private-interface theorem is a stylized query-complexity model, not a security proof for a deployed auditor. Third, the image evidence is primarily CLIP-centered; the additional ResNet-18 result covers one victim, one seed, one score interface, and partial transfer in one direction. All image experiments use same-prompt real/diffusion benchmarks and bounded pixel edits. Semantic edits, non-diffusion sources, and live commercial APIs are not evaluated. Finally, trusted marks, keys, signatures, or authenticated metadata require a separate provenance threat model; no active or hybrid system is evaluated here.

The deployment lesson is that clean provenance accuracy is not a robustness certificate. A passive verifier must be evaluated jointly with the interface through which it is exposed. In practice, this means that a robust evaluation should state three objects together: the source–target direction, the edit class, and the released interface. Omitting any one of these can invert the conclusion. A detector may look strong when averaged over a binary human-versus-AI task, weak when the same images are evaluated as generator impersonation, and weaker still when the public interface reveals scores that make the verifier easy to emulate. In the evaluated CLIP systems, abstention improves the attack-derived gap estimate without changing the source distributions. This observation does not establish the sign of the true robust gap or a change in the statistical ceiling.

9 Conclusion

We formulated passive image provenance as source–target verification under adversarial distribution shift. The resulting minimax game has an exact value: the best robust gap achievable by any image-only verifier is the total-variation distance from the target distribution to the attacked source family. We then showed a separate public-interface barrier: once a verifier is learnable on the attack region, surrogate black-box attacks approach white-box performance. A finite-state calculation checks the minimax identity. On the same-prompt image benchmarks, the evaluated public CLIP verifiers fail under targeted pixel attacks; a ResNet-18 victim shows partial fake-to-real transfer, and paired metrics quantify the image-quality trade-off. Abstention reduces measured attack success in the CLIP comparison, while positive estimated gap upper bounds leave robustness unresolved. Robust passive provenance therefore requires both statistical source–target analysis and interface-aware evaluation. The central recommendation is to report these two axes separately: estimate or bracket the image-level source–target separation where possible, and stress-test the exact interface that users or adversaries will observe. This framing also makes negative results actionable: a failure caused by a small distributional radius calls for stronger provenance signals or active marks, whereas a failure caused by a learnable interface calls for interface redesign and a renewed black-box audit.

Acknowledgments and Disclosure of Funding

We thank the anonymous reviewers for their valuable feedback, and the Edinburgh International Data Facility (EIDF) and the Data-Driven Innovation Programme at the University of Edinburgh for computing resources. This work received no specific grant funding. The author declares no competing interests.

References

[1] Jan Lukáš, Jessica Fridrich, and Miroslav Goljan. 2006. Digital camera identification from sensor pattern noise. *IEEE Transactions on Information Forensics and Security* 1, 2 (2006),

205–214.

[2] Davide Cozzolino and Luisa Verdoliva. 2020. Noiseprint: A CNN-based camera model fingerprint. *IEEE Transactions on Information Forensics and Security* 15 (2020), 144–159.

[3] Davide Cozzolino, Justus Thies, Andreas Rössler, Christian Riess, Matthias Nießner, and Luisa Verdoliva. 2018. ForensicTransfer: Weakly-supervised domain adaptation for forgery detection. *arXiv preprint arXiv:1812.02510*.

[4] Sheng-Yu Wang, Oliver Wang, Richard Zhang, Andrew Owens, and Alexei A. Efros. 2020. CNN-generated images are surprisingly easy to spot... for now. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 8695–8704.

[5] Utkarsh Ojha, Yuheng Li, and Yong Jae Lee. 2023. Towards universal fake image detectors that generalize across generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 24480–24489.

[6] Zhendong Wang, Jianmin Bao, Wengang Zhou, Weilun Wang, Hezhen Hu, Hong Chen, and Houqiang Li. 2023. DIRE for diffusion-generated image detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 22445–22455.

[7] Davide Cozzolino, Giovanni Poggi, Riccardo Corvi, Matthias Nießner, and Luisa Verdoliva. 2024. Raising the bar of AI-generated image detection with CLIP. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. 4356–4366.

[8] Ning Yu, Larry S. Davis, and Mario Fritz. 2019. Attributing fake images to GANs: Learning and analyzing GAN fingerprints. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 7556–7566.

[9] Ning Yu, Vladislav Skripniuk, Dingfan Chen, Larry Davis, and Mario Fritz. 2022. Responsible disclosure of generative models using scalable fingerprinting. In *International Conference on Learning Representations (ICLR)*.

[10] Tianyun Yang, Danding Wang, Fan Tang, Xinying Zhao, Juan Cao, and Sheng Tang. 2023. Progressive open space expansion for open-set model attribution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 15856–15865.

[11] Hae Jin Song, Mahyar Khayatkhoei, and Wael AbdAlmageed. 2024. ManiFPT: Defining and analyzing fingerprints of generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 10791–10801.

[12] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. 2015. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations (ICLR)*.

[13] Nicolas Papernot, Patrick McDaniel, Ian Goodfellow, Somesh Jha, Z. Berkay Celik, and Ananthram Swami. 2017. Practical black-box attacks against machine learning. In *Proceedings of the ACM Asia Conference on Computer and Communications Security (AsiaCCS)*. 506–519.

[14] Nicholas Carlini and David Wagner. 2017. Towards evaluating the robustness of neural networks. In *IEEE Symposium on Security and Privacy (SP)*. 39–57.

[15] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. 2018. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations (ICLR)*.

[16] Anish Athalye, Nicholas Carlini, and David Wagner. 2018. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*. 274–283.

[17] Nicholas Carlini and Hany Farid. 2020. Evading deepfake-image detectors with white- and black-box attacks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. 2804–2813.

[18] Shehzeen Hussain, Paarth Neekhara, Malhar Jere, Farinaz Koushanfar, and Julian McAuley. 2021. Adversarial deepfakes: Evaluating vulnerability of deepfake detectors to adversarial examples. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. 3348–3357.

[19] Vera Wesselkamp, Konrad Rieck, Daniel Arp, and Erwin Quiring. 2022. Misleading deep-fake detection with GAN fingerprints. In *2022 IEEE Security and Privacy Workshops (SPW), Deep Learning and Security Workshop (DLS)*. 59–65.

[20] Kai Yao and Marc Juarez. 2026. Smudged Fingerprints: A Systematic Evaluation of the Robustness of AI Image Fingerprints. In *Proceedings of the 2026 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*. Also available as arXiv:2512.11771.

[21] Andreas Rössler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner. 2019. FaceForensics++: Learning to detect manipulated facial images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 1–11.

[22] Hao Dang, Feng Liu, Joel Stehouwer, Xiaoming Liu, and Anil K. Jain. 2020. On the detection of digital face manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 5781–5790.

[23] Lingzhi Li, Jianmin Bao, Ting Zhang, Hao Yang, Dong Chen, Fang Wen, and Baining Guo. 2020. Face X-Ray for more general face forgery detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 5001–5010.

[24] Kaede Shiohara and Toshihiko Yamasaki. 2022. Detecting deepfakes with self-blended images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 18720–18729.

[25] Tarik Dzanic, Karan Shah, and Freddie Witherden. 2020. Fourier spectrum discrepancies in deep network generated images. In *Advances in Neural Information Processing Systems (NeurIPS)* 33, 3022–3032.

[26] Keshigeyan Chandrasegaran, Ngoc-Trung Tran, and Ngai-Man Cheung. 2021. A closer look at Fourier spectrum discrepancies for CNN-generated images detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 7200–7209.

[27] Jeongsoo Park and Andrew Owens. 2025. Community Forensics: Using thousands of generators to train fake image detectors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 8245–8257.

[28] Dimitrios Karageorgiou, Symeon Papadopoulos, Ioannis Kompatsiaris, and Efstratios Gavves. 2025. Any-resolution AI-generated image detection by spectral learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 18706–18717.

[29] Zhiyuan Yan, Yong Zhang, Xinhang Yuan, Siwei Lyu, and Baoyuan Wu. 2023. DeepfakeBench: A comprehensive benchmark of deepfake detection. In *Advances in Neural Information Processing Systems (NeurIPS)* 36, 4534–4565.

[30] Zhiyuan Yan, Taiping Yao, Shen Chen, Yandan Zhao, Xinghe Fu, Junwei Zhu, Donghao Luo, Chengjie Wang, Shouhong Ding, Yunsheng Wu, and Li Yuan. 2024. DF40: Toward next-generation deepfake detection. In *Advances in Neural Information Processing Systems (NeurIPS)* 37, 29387–29434. Datasets and Benchmarks Track.

[31] Arpan Mahara and Naphtali Rishe. 2026. Methods and trends in detecting AI-generated images: A comprehensive review. *Computer Science Review* 60, 100908.

[32] Ning Yu, Vladislav Skripniuk, Sahar Abdelnabi, and Mario Fritz. 2021. Artificial fingerprinting for generative models: Rooting deepfake attribution in training data. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 14448–14457.

[33] Sharath Girish, Saksham Suri, Sai Saketh Rambhatla, and Abhinav Shrivastava. 2021. Towards discovery and attribution of open-world GAN generated images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 14094–14103.

[34] Luca Guarnera, Oliver Giudice, Matthias Nießner, and Sebastiano Battiato. 2022. On the exploitation of deepfake model recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. 61–70.

[35] Tianyun Yang, Ziyao Huang, Juan Cao, Lei Li, and Xirong Li. 2022. Deepfake network architecture attribution. In *Proceedings of the AAAI Conference on Artificial Intelligence* 36, 4 (2022), 4662–4670.

[36] Zhimin Sun, Shen Chen, Taiping Yao, Bangjie Yin, Ran Yi, Shouhong Ding, and Lizhuang Ma. 2023. Contrastive pseudo learning for open-world DeepFake attribution. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 20882–20892.

[37] Hae Jin Song and Laurent Itti. 2025. Riemannian-geometric fingerprints of generative models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 11425–11435.

[38] Tai D. Nguyen, Aref Azizpour, and Matthew C. Stamm. 2025. Forensic self-descriptions are all you need for zero-shot detection, open-set source attribution, and clustering of AI-generated images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 3040–3050.

[39] Yunfeng Diao, Naixin Zhai, Changtao Miao, Zitong Yu, Xingxing Wei, Xun Yang, and Meng Wang. 2026. Vulnerabilities in AI-Generated Image Detection: The Challenge of Adversarial Attacks. *IEEE Transactions on Multimedia*, early access, 1–13. `https://doi.org/10.1109/TMM.2026.3682154`.

[40] Pierre Fernandez, Guillaume Couairon, Hervé Jégou, Matthijs Douze, and Teddy Furon. 2023. The Stable Signature: Rooting watermarks in latent diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 22466–22477.

[41] Yuxin Wen, John Kirchenbauer, Jonas Geiping, and Tom Goldstein. 2023. Tree-Rings Watermarks: Invisible Fingerprints for Diffusion Images. In *Advances in Neural Information Processing Systems (NeurIPS)* 36, 58047–58063.

[42] Changhoon Kim, Kyle Min, Maitreya Patel, Sheng Cheng, and Yezhou Yang. 2024. WOUAF: Weight modulation for user attribution and fingerprinting in text-to-image diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 8974–8983.

[43] Bang An, Mucong Ding, Tahseen Rabbani, Aakriti Agrawal, Yuancheng Xu, Chenghao Deng, Sicheng Zhu, Abdirisak Mohamed, Yuxin Wen, Tom Goldstein, and Furong Huang. 2024. WAVES: Benchmarking the robustness of image watermarks. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, PMLR 235, 1456–1492.

[44] Rui Xu, Mengya Hu, Deren Lei, Yaxi Li, David Lowe, Alex Gorevski, Mingyu Wang, Emily Ching, and Alex Deng. 2025. InvisMark: Invisible and robust watermarking for AI-generated image provenance. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. 909–918.

[45] Hanlin Zhang, Benjamin L. Edelman, Danilo Francati, Daniele Venturi, Giuseppe Ateniese, and Boaz Barak. 2024. Watermarks in the Sand: Impossibility of Strong Watermarking for Language Models. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, PMLR 235, 58851–58880.

[46] Jaiden Fairoze, Guillermo Ortiz-Jiménez, Mel Vecerik, Somesh Jha, and Sven Gowal. 2025. On the difficulty of constructing a robust and publicly-detectable watermark. In *Proceedings of the 28th International Conference on Artificial Intelligence and Statistics (AISTATS)*, PMLR 258, 1891–1899.

[47] Kai Yao and Marc Juarez. 2026. SPRINT: Robust Model Attribution of Generated Images via Secret Pixel Reconstruction. *arXiv preprint arXiv:2508.05691v3*.

[48] Maurice Sion. 1958. On general minimax theorems. *Pacific Journal of Mathematics* 8, 1 (1958), 171–176.

[49] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. 2014. Microsoft COCO: Common objects in context. In *European Conference on Computer Vision (ECCV)*. 740–755.

[50] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 10684–10695.

[51] 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. 2021. Learning Transferable Visual Models From Natural Language Supervision. In *Proceedings of the 38th International Conference on Machine Learning (ICML)*, PMLR 139, 8748–8763.

[52] J. C. Neves, R. Tolosana, R. Vera-Rodriguez, et al. 2020. GANprintR: Improved fakes and evaluation of the state of the art in face manipulation detection. *IEEE Journal of Selected Topics in Signal Processing* 14, 5 (2020), 1038–1048.

[53] Sina Mavali, Jonas Ricker, David Pape, Asja Fischer, and Lea Schönherr. 2026. Adversarial Robustness of AI-Generated Image Detectors in the Real World. Accepted at the *23rd International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment (DIMVA)*. *arXiv:2410.01574v4*.

[54] Zhenting Wang, Chen Chen, Yi Zeng, Lingjuan Lyu, and Shiqing Ma. 2023. Where Did I Come From? Origin Attribution of AI-Generated Images. In *Advances in Neural Information Processing Systems (NeurIPS)* 36, 74478–74500.

[55] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 770–778.

[56] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. 2018. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 586–595.

[57] Zhou Wang, Alan C. Bovik, Hamid R. Sheikh, and Eero P. Simoncelli. 2004. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing* 13, 4, 600–612.

[58] Florian Tramèr, Fan Zhang, Ari Juels, Michael K. Reiter, and Thomas Ristenpart. 2016. Stealing machine learning models via prediction APIs. In *25th USENIX Security Symposium (USENIX Security)*. 601–618.

[59] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

A Additional Formal Details for Barrier I

A.1 Reachable attacked distributions

Lemma 10 (Reachable sets are convex and compact)**.** *For every source label s, the set $\mathcal{Q}_s$ is nonempty, convex, and compact.*

Proof. Let

$$\mathcal{K}_\Gamma := \left\{ A(\cdot \mid x)_{x\in\mathcal{X}} : \begin{array}{l} A(\cdot \mid x) \text{ is a distribution} \\ \text{supported on } \Gamma(x) \text{ for every } x \in \mathcal{X} \end{array} \right\}.$$

Because $\mathcal{X}$ is finite and each $\Gamma(x)$ is nonempty, $\mathcal{K}_\Gamma$ is a nonempty product of simplices, hence compact and convex. The map

$$\Psi : \mathcal{K}_\Gamma \to \mathcal{P}(\mathcal{X}), \qquad \Psi(A) = A_{\#}P_s,$$

is linear and continuous. Therefore $\mathcal{Q}_s = \Psi(\mathcal{K}_\Gamma)$ is nonempty, compact, and convex. □

Proposition 11 (White-box optimum is pointwise)**.** *For every target verifier $f_t : \mathcal{X} \to [0,1]$,*

$$W_{s\to t}(f_t;\Gamma) = \mathbb{E}_{X\sim P_s}\Big[\sup_{z\in\Gamma(X)} f_t(z)\Big].$$

In particular, randomized attacks do not outperform a pointwise maximizer.

Proof. For any admissible attack kernel A and each $x \in \mathcal{X}$,

$$\mathbb{E}_{Z\sim A(\cdot|x)}[f_t(Z)] \le \sup_{z\in\Gamma(x)} f_t(z).$$

Averaging over $X \sim P_s$ gives the upper bound. Since $\mathcal{X}$ is finite and $\Gamma(x)$ is nonempty, for each x there exists a maximizer $a_t(x) \in \arg\max_{z\in\Gamma(x)} f_t(z)$. The deterministic kernel $A_t(\cdot \mid x) = \delta_{a_t(x)}$ attains equality. □

A.2 Total variation preliminaries

Lemma 12 (Variational form of total variation)**.** *For any two distributions P, Q on $\mathcal{X}$,*

$$\mathrm{TV}(P,Q) = \sup_{f:\mathcal{X}\to[0,1]} \big(\mathbb{E}_P[f] - \mathbb{E}_Q[f]\big).$$

Proof. For any event $B \subseteq \mathcal{X}$, the indicator $\mathbf{1}_B$ belongs to $[0,1]^{\mathcal{X}}$, so

$$\sup_{f:\mathcal{X}\to[0,1]} (\mathbb{E}_P[f] - \mathbb{E}_Q[f]) \ge \sup_{B\subseteq\mathcal{X}} (P(B) - Q(B)) = \mathrm{TV}(P,Q).$$

Conversely, for any $f : \mathcal{X} \to [0,1]$,

$$\mathbb{E}_P[f] - \mathbb{E}_Q[f] = \sum_{x\in\mathcal{X}} f(x)(P(x) - Q(x)) \le \sum_{x:P(x)\ge Q(x)} (P(x) - Q(x)) = \mathrm{TV}(P,Q).$$

Taking the supremum proves equality. □

Proposition 13 (Best verifier against a fixed attacked distribution)**.** *Fix a target distribution P_t and any comparison distribution Q on $\mathcal{X}$. Define*

$$A_Q := \{x \in \mathcal{X} : P_t(x) \ge Q(x)\}, \qquad f_Q := \mathbf{1}_{A_Q}.$$

Then

$$\mathbb{E}_{P_t}[f_Q] - \mathbb{E}_Q[f_Q] = \mathrm{TV}(P_t, Q).$$

Proof. By definition of A_Q,

$$\mathbb{E}_{P_t}[f_Q] - \mathbb{E}_Q[f_Q] = P_t(A_Q) - Q(A_Q) = \sum_{x\in A_Q} (P_t(x) - Q(x)).$$

Because $P_t(x) - Q(x) \ge 0$ exactly on A_Q and the total positive mass equals the total variation distance, the claim follows. □

A.3 Full proof of the attackability-radius law

Proof of Theorem 1. For every verifier f_t,

$$\begin{aligned}
\mathrm{Gap}_{s,t}(f_t) &= \mathrm{Acc}_t(f_t) - W_{s\to t}(f_t;\Gamma) \\
&= \mathbb{E}_{P_t}[f_t] - \sup_{Q\in\mathcal{Q}_s} \mathbb{E}_Q[f_t] \\
&= \inf_{Q\in\mathcal{Q}_s} \left(\mathbb{E}_{P_t}[f_t] - \mathbb{E}_Q[f_t]\right).
\end{aligned}$$

Let

$$L(f_t, Q) := \mathbb{E}_{P_t}[f_t] - \mathbb{E}_Q[f_t].$$

By Lemma 10, $\mathcal{Q}_s$ is nonempty, convex, and compact. Since $\mathcal{X}$ is finite, $[0,1]^{\mathcal{X}}$ is also convex and compact. The payoff L is affine in f_t and affine in Q. Sion's minimax theorem therefore applies:

$$\sup_{f_t} \inf_{Q \in \mathcal{Q}_s} L(f_t, Q) = \inf_{Q \in \mathcal{Q}_s} \sup_{f_t} L(f_t, Q).$$

For any fixed attacked distribution Q, Lemma 12 gives

$$\sup_{f_t : \mathcal{X} \to [0,1]} L(f_t, Q) = \mathrm{TV}(P_t, Q).$$

Combining these identities yields

$$\sup_{f_t : \mathcal{X} \to [0,1]} \mathrm{Gap}_{s,t}(f_t) = \inf_{Q \in \mathcal{Q}_s} \mathrm{TV}(P_t, Q) = \Delta_{s,t}.$$

□

A.4 Coarsening and directional asymmetry

Proof of Proposition 3. By compactness of $\mathcal{Q}_s$, for each j there exists

$$Q_j^\star \in \arg\min_{Q \in \mathcal{Q}_s} \mathrm{TV}(P_j, Q)$$

with $\mathrm{TV}(P_j, Q_j^\star) = \Delta_{s,j}$. Define $\overline{Q} := \sum_j \pi_j Q_j^\star$. By convexity, $\overline{Q} \in \mathcal{Q}_s$. Joint convexity of total variation gives

$$\mathrm{TV}(P_{\mathsf{A}}, \overline{Q}) = \mathrm{TV}\Big(\sum_j \pi_j P_j, \sum_j \pi_j Q_j^\star \Big) \le \sum_j \pi_j \mathrm{TV}(P_j, Q_j^\star) = \sum_j \pi_j \Delta_{s,j}.$$

Since $\Delta_{s,\mathsf{A}} = \inf_{Q \in \mathcal{Q}_s} \mathrm{TV}(P_{\mathsf{A}}, Q) \le \mathrm{TV}(P_{\mathsf{A}}, \overline{Q})$, the result follows. □

Proposition 14 (No universal ordering between human evasion and model impersonation)**.** *There is no universal inequality relating $\Delta_{s,\mathsf{H}}$ and $\Delta_{s,t}$ for a specific model target t. Both strict inequalities can occur.*

Proof. We give two finite-space examples. Let $\mathcal{X} = \{a, b, c\}$, $P_s = \delta_a$, and $\Gamma(a) = \{a, b\}$.

First let $P_{\mathsf{H}} = \frac{1}{2}\delta_b + \frac{1}{2}\delta_c$ and $P_t = \delta_c$. Since every reachable attacked distribution is supported on $\{a, b\}$, choosing $Q = \delta_b$ gives $\Delta_{s,\mathsf{H}} = 1/2$, while every reachable Q is disjoint from δ_c, so $\Delta_{s,t} = 1$.

Second let $P_{\mathsf{H}} = \delta_c$ and $P_t = \delta_b$. Then choosing $Q = \delta_b$ gives $\Delta_{s,t} = 0$, while every reachable Q is disjoint from δ_c, so $\Delta_{s,\mathsf{H}} = 1$. □

B Additional Formal Details for Barrier II

B.1 Proof of the surrogate-attack theorem

Proof of Theorem 4. Fix $x \in \mathcal{X}$ and let

$$z_x^\star \in \arg\max_{z \in \Gamma(x)} f_t(z).$$

Since $\widehat{f}_t$ is an ε-emulator on $\mathcal{U}$ and $z_x^\star \in \Gamma(x) \subseteq \mathcal{U}$,

$$\widehat{f}_t(z_x^\star) \ge f_t(z_x^\star) - \varepsilon = \sup_{z \in \Gamma(x)} f_t(z) - \varepsilon.$$

By definition of $\alpha(x)$,

$$\widehat{f}_t(\widehat{a}(x)) \ge \sup_{z \in \Gamma(x)} \widehat{f}_t(z) - \alpha(x) \ge \widehat{f}_t(z_x^\star) - \alpha(x).$$

Applying the emulator bound again at $\widehat{a}(x)$ gives

$$\begin{aligned} f_t(\widehat{a}(x)) &\ge \widehat{f}_t(\widehat{a}(x)) - \varepsilon \\ &\ge \sup_{z \in \Gamma(x)} f_t(z) - 2\varepsilon - \alpha(x). \end{aligned}$$

Averaging over $X \sim P_s$ and using Proposition 11,

$$\begin{aligned}\mathbb{E}_{X\sim P_s}[f_t(\widehat{a}(X))] &\geq \mathbb{E}_{X\sim P_s}\left[\sup_{z\in\Gamma(X)} f_t(z)\right] - 2\varepsilon - \bar{\alpha} \\ &= W_{s\to t}(f_t;\Gamma) - 2\varepsilon - \bar{\alpha}.\end{aligned}$$

□

Proof of Corollary 5. On the event $\{\|\widehat{f}_t - f_t\|_{\infty,\mathcal{U}} \leq \varepsilon\}$, Theorem 4 applies. This event has probability at least $1-\delta$ by assumption. □

B.2 Exact recovery of score-revealing heads

Proof of Proposition 6. For the softmax head, fix the gauge $h_m = 0$. For each probe point x_i and each class $j < m$,

$$\log \frac{g_H(j \mid x_i)}{g_H(m \mid x_i)} = \langle h_j, \phi(x_i)\rangle - \langle h_m, \phi(x_i)\rangle = \langle h_j, \phi(x_i)\rangle.$$

Stacking the p equations for a fixed class j gives

$$\Phi h_j = \begin{bmatrix} \log \frac{g_H(j|x_1)}{g_H(m|x_1)} \\ \vdots \\ \log \frac{g_H(j|x_p)}{g_H(m|x_p)} \end{bmatrix}.$$

Since Φ is invertible, each h_j is exactly determined. The gauge determines $h_m = 0$, so the whole head H is recovered. The binary logistic case is obtained by taking $m = 2$ and converting returned probabilities into logits.

Exact recovery gives an exact emulator of every target coordinate $f_t(x) = g_H(t \mid x)$. Applying Theorem 4 with $\varepsilon = 0$ shows that attacks on the recovered head match white-box attacks up to optimization error. □

Proof of Proposition 7. For each class $j < m$, let $e_j \in \mathbb{R}^p$ be the vector of log-odds errors at the probe points. By assumption, $\|e_j\|_2 \leq \sqrt{p}\,\eta$. The recovered vector satisfies

$$\Phi\widehat{h}_j = \Phi h_j + e_j,$$

so

$$\widehat{h}_j - h_j = \Phi^{-1} e_j.$$

Therefore

$$\|\widehat{h}_j - h_j\|_2 \leq \|\Phi^{-1}\|_2 \|e_j\|_2 \leq \|\Phi^{-1}\|_2 \sqrt{p}\,\eta.$$

For any $u \in \mathcal{U}$ with $\|\phi(u)\|_2 \leq R$,

$$|\langle \widehat{h}_j - h_j, \phi(u)\rangle| \leq \|\widehat{h}_j - h_j\|_2 \|\phi(u)\|_2 \leq R\|\Phi^{-1}\|_2 \sqrt{p}\,\eta.$$

Let $\ell(u)$ and $\widehat{\ell}(u)$ denote the true and recovered logit vectors in the same gauge, with $\ell_m(u) = \widehat{\ell}_m(u) = 0$. The preceding per-class bound gives $\|\widehat{\ell}(u) - \ell(u)\|_\infty \leq B$ uniformly on $\mathcal{U}$. For a softmax target coordinate $s_t(\ell)$,

$$\|\nabla s_t(\ell)\|_1 = 2s_t(\ell)(1 - s_t(\ell)) \leq \tfrac{1}{2}.$$

The mean-value bound therefore gives $\|\widehat{f}_t - f_t\|_{\infty,\mathcal{U}} \leq B/2$. In the binary reference-class gauge, the target probability is a logistic function of one log-odds value; its derivative is at most $1/4$, giving the sharper bound $B/4$. □

C Private-Interface Counting Proofs

Proof of Theorem 8. Condition on the attacker's internal randomness, yielding a deterministic adaptive strategy. There are at most M^q possible transcripts. Each transcript determines one final action a. By assumption, the set of hidden states on which any fixed action succeeds has size at most $\rho|\Theta|$. If the attacker succeeds, the realized hidden state must lie in the success set of the action corresponding to the realized transcript. Thus the success set is contained in a union of at most M^q sets, each of size at most $\rho|\Theta|$. The success probability is therefore at most $M^q\rho$. Averaging over the attacker's randomness preserves the bound. □

C.1 Hidden-support instantiation

Fix integers d, k, b, τ with $1 \le \tau \le b \le d$ and $\tau \le k \le d$. A verifier secretly samples a support $S \subseteq [d]$, $|S| = k$, uniformly from all k-subsets. A candidate forged sample is represented by a set $B \subseteq [d]$ of activated coordinates with $|B| \le b$. The verifier accepts if and only if $|B \cap S| \ge \tau$.

For a fixed set B with $|B| = b$, define

$$N_{d,k,b,\tau} := \sum_{i=\tau}^{\min\{k,b\}} \binom{b}{i}\binom{d-b}{k-i}.$$

Then

$$\Pr[|S \cap B| \ge \tau] = \frac{N_{d,k,b,\tau}}{\binom{d}{k}},$$

because achieving intersection size i requires choosing i elements from B and $k - i$ from its complement. Applying Theorem 8 with $M = 2$ and

$$\rho = \frac{N_{d,k,b,\tau}}{\binom{d}{k}}$$

gives Corollary 9. If $|B| < b$, enlarge it to any superset of size b, which can only increase its acceptance probability.

For the simpler upper bound, the event $|S \cap B| \ge \tau$ implies that some τ-subset $T \subseteq B$ lies inside S. By a union bound,

$$\Pr[|S \cap B| \ge \tau] \le \sum_{\substack{T \subseteq B \\ |T|=\tau}} \Pr[T \subseteq S].$$

For each fixed T,

$$\Pr[T \subseteq S] = \frac{\binom{d-\tau}{k-\tau}}{\binom{d}{k}} = \prod_{r=0}^{\tau-1} \frac{k-r}{d-r} \le \left(\frac{k}{d}\right)^{\tau}.$$

Hence

$$\Pr[|S \cap B| \ge \tau] \le \binom{b}{\tau}\left(\frac{k}{d}\right)^{\tau}.$$

Multiplying by 2^q gives the q-query bound.

Corollary 15 (Explicit sparse-support query lower bound)**.** *Assume $ebk/(\tau d) < 1$. Any binary-query attacker that succeeds with probability at least η in the hidden-support threshold model must make at least*

$$q \ge \tau \log_2\left(\frac{\tau d}{ebk}\right) - \log_2\left(\frac{1}{\eta}\right)$$

queries.

Proof. If success probability is at least η, then the preceding bound gives

$$\eta \le 2^q \left(\frac{ebk}{\tau d}\right)^{\tau},$$

using $\binom{b}{\tau} \le (eb/\tau)^{\tau}$. Rearranging proves the claim. □

D Additional Empirical Results

This appendix is the empirical audit trail for Section 7. It is organized around the three empirical claims made in the main text rather than as a collection of independent tables. First, Section D.1 verifies the public-head recovery mechanism used in Section 7.1: score access lets the attacker identify the deployed head and then attack it nearly as a white-box model. Second, Section D.2 supports Section 7.2: changing the released interface changes attackability more than changing clean accuracy alone. Third, Section D.3 supports Section 7.3: stronger clean training and adversarial retraining retain negative estimated gap upper bounds in the evaluated public setting.

Unless stated otherwise, interface results are reported at the representative $8/255$ slice used in the main text, and $\widetilde{\mathrm{Gap}}$ is the empirical upper bound defined in Section 7. A table is included only when it checks a specific link in the main argument: recovery quality, directional collapse, clean operating point, interface bandwidth, transcript caps, score-guided random search, or retraining controls.

D.1 Controlled public CLIP-softmax baseline

The controlled three-class experiment in Section 7.1 uses a public CLIP-softmax verifier. The purpose of Table 3 is to connect Proposition 6 to the measured attack pipeline. The first numerical column tracks score emulation error, the second tracks the gap between attacks optimized on the recovered model and direct white-box attacks, and the final column checks whether the recovered model is already sufficient for targeted success. Thus the table is not merely a query sweep: it shows the transition from approximate recovery to operationally white-box behavior.

Table 3: Exact-recovery details for the controlled public CLIP-softmax baseline on the three-class benchmark. The table supports Section 7.1 by showing that score emulation becomes operationally white-box by 2,048 queries.

Queries	Score error	Rec.-WB gap	Recovered ASR
128	3.03×10^{-3}	0.0363	1.000
256	1.89×10^{-3}	0.0192	1.000
512	1.01×10^{-3}	0.0162	1.000
1,024	1.26×10^{-3}	0.0052	1.000
2,048	6.88×10^{-8}	0.0030	1.000
7,200	5.60×10^{-8}	0.0028	1.000

Once the public head is recovered, the next question is whether collapse is confined to one source–target direction. Table 4 answers that question. It expands the aggregate statement in Figure 1: human false-flagging, AI evasion, and generator impersonation all reach ASR 1.0 by the largest evaluated budget, and every reported $\widetilde{\mathrm{Gap}}$ is negative. The last two columns make the negative-gap interpretation explicit on the held-out samples: attacked source images receive higher average target score than the clean target images themselves.

Table 4: Per-direction targeted attacks for the original controlled public CLIP-softmax verifier on the three-class benchmark. This table supports the negative-gap claim in Section 7.1: all four source–target directions reach ASR 1.0 by $8/255$, and every empirical gap upper bound is negative.

Direction	Budget	ASR	$\widetilde{\mathrm{Gap}}$	Clean target acc.	Adv. target score
Human→AI	$2/255$	1.0000	-0.0387	0.6786	0.7173
Human→AI	$4/255$	1.0000	-0.0522	0.6786	0.7308
Human→AI	$8/255$	1.0000	-0.0628	0.6786	0.7414
AI→Human	$2/255$	1.0000	-0.0420	0.3338	0.3758
AI→Human	$4/255$	1.0000	-0.0592	0.3338	0.3930
AI→Human	$8/255$	1.0000	-0.0755	0.3338	0.4094
SD1.5→SDXL	$2/255$	1.0000	-0.0420	0.3590	0.4010
SD1.5→SDXL	$4/255$	1.0000	-0.0636	0.3590	0.4226
SD1.5→SDXL	$8/255$	1.0000	-0.0805	0.3590	0.4395
SDXL→SD1.5	$2/255$	0.8900	-0.0260	0.3345	0.3605
SDXL→SD1.5	$4/255$	0.9800	-0.0394	0.3345	0.3739
SDXL→SD1.5	$8/255$	1.0000	-0.0540	0.3345	0.3885

The final check for the controlled public model is whether coarsening to a binary AI target avoids the attribution-style impersonation failure discussed after Proposition 3. It does not in this setting. At $8/255$, all evaluated human→AI mixture attacks—both oracle-mixture and coarse-only, with mixture weight $\pi \in \{0, 0.25, 0.5, 0.75, 1\}$—finish with ASR 1.0. This observation is used only as an operational check; the formal coarsening statement remains the distributional inequality in Proposition 3.

D.2 Expanded five-class interface study

The expanded benchmark adds more generators and interface variants. Table 5 first establishes the clean operating point. Its role is to prevent a misleading reading of Table 1: the selected linear, nonlinear, and off-the-shelf CLIP systems are not arbitrary weak detectors, while CNNSpot and DIRE are retained only as weak negative controls on this same-prompt benchmark. The main interface conclusions are therefore drawn from the three representative systems used in Table 1, not from the weak controls.

Table 5: Clean matrix on the expanded five-class benchmark. This table supports the setup of Section 7.2: the three representative CLIP-based systems have meaningful clean performance, while CNNSpot and DIRE are included as weak off-the-shelf controls rather than as main robustness claims.

Verifier	Acc.	Macro-F1	Fake AUC	TPR@5%FPR	EER	Role
Binary logistic on CLIP	0.8007	0.4479	0.8470	0.4033	0.2288	Shared-representation linear baseline
Intercept softmax on CLIP	0.5053	0.4782	0.9054	0.5217	0.1608	Shared-representation linear variant
One-vs-rest logistic on CLIP	0.5980	0.5871	0.9338	0.6433	0.1371	Shared-representation linear variant
Temperature-scaled softmax on CLIP	0.4827	0.4491	0.9286	0.7133	0.1525	Shared-representation linear variant
Linear softmax head on CLIP	0.5667	0.5734	0.9289	0.6608	0.1438	Main-text linear representative
Two-layer MLP on CLIP	0.6280	0.6071	0.9717	0.8333	0.0883	Main-text nonlinear representative
CLIP-based off-the-shelf detector	0.6773	0.6082	0.7231	0.2475	0.3367	Main-text off-the-shelf representative
CNNSpot	0.2013	0.1685	0.5108	0.0292	0.4804	Weak off-the-shelf control
DIRE	0.1900	0.1740	0.3090	0.0083	0.6554	Weak off-the-shelf control

With the clean operating point fixed, Table 6 is the central interface table behind Table 1. Each row changes what the attacker sees while holding the same benchmark, target direction family, attack budget, and representative verifiers fixed. The ASR columns show attack success, the $\widetilde{\mathrm{Gap}}$ columns estimate attack-derived upper bounds rather than certify positive robust separation, and the ϵ_{95} columns report validation-set emulator diagnostics. The important pattern is that high emulator error is not sufficient by itself: top-1 feedback has $\epsilon_{95} = 1.0$ but can still be attacked, whereas binary feedback with abstention both removes the smooth target-aligned score and rejects borderline cases.

Table 6: Representative interface slice at $8/255$ on the expanded five-class benchmark. This table is the detailed version of Table 1: columns report ASR, the empirical gap upper bound $\widetilde{\mathrm{Gap}}$, and the 95th-percentile emulator error ϵ_{95}. Here q is a per-split transcript cap, yielding $\min(q, 3000)$ training and $\min(q, 1500)$ validation replies. Dashes denote model–interface combinations not evaluated in this sweep.

	Linear softmax on CLIP			**Two-layer MLP on CLIP**			**Off-the-shelf CLIP detector**		
Interface	**ASR**	$\widetilde{\mathrm{Gap}}$	ϵ_{95}	**ASR**	$\widetilde{\mathrm{Gap}}$	ϵ_{95}	**ASR**	$\widetilde{\mathrm{Gap}}$	ϵ_{95}
Full score vector	1.0000	−0.0356	0.0051	1.0000	−0.0598	0.0054	1.0000	−0.5515	0.2824
Single fake score	0.9100	−0.0098	0.3314	0.9300	−0.0200	0.3137	1.0000	−0.5405	0.3476
8-bit fake score	0.9233	−0.0100	0.3196	0.9733	−0.0209	0.3313	–	–	–
Top-1 label only	0.7533	−0.0076	1.0000	1.0000	−0.0380	1.0000	1.0000	−0.5512	1.0000
Binary verdict	1.0000	−0.0260	0.7056	1.0000	−0.0498	0.7004	–	–	–
Binary + abstain ($q = 128$)	0.0000	+0.0137	1.0000	0.0000	+0.0104	1.0000	0.0300	+0.2858	1.0000
Binary + abstain ($q = 7{,}200$)	0.0000	+0.0104	1.0000	0.0000	+0.0120	1.0000	0.0933	+0.2165	1.0000

Figure 3 gives the same comparison as a visual diagnostic. It is included to make the three-way pattern easier to inspect: emulator error, ASR, and $\widetilde{\mathrm{Gap}}$ move differently across interfaces, so a single metric would obscure the reason that abstention helps.

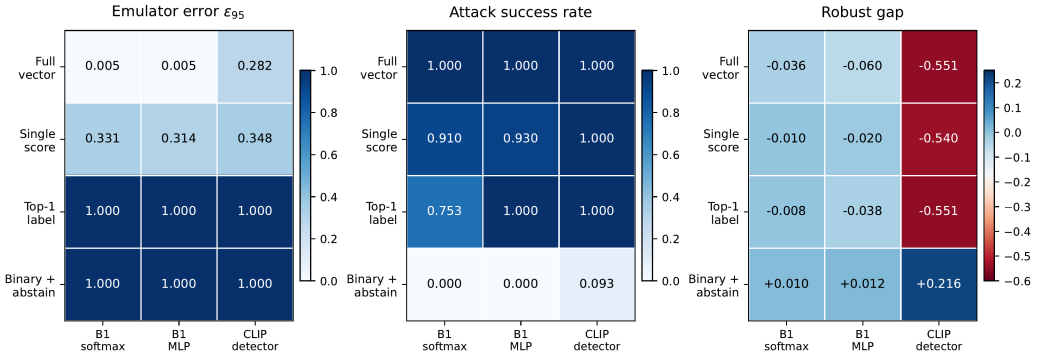


Figure 3: Heatmap summary of the representative $8/255$ interface slice. This visual counterpart to Table 6 shows that richer interfaces remain fragile and that positive empirical gap upper bounds first appear only under binary feedback with abstention.

Table 7 varies the availability of binary transcripts without abstention. ASR remains 1.0 and estimated gap upper bounds remain negative across the tested per-split caps. These are controlled transcript-availability measurements, not end-to-end commercial API query budgets. They show that ordinary binary feedback remains attackable under the tested caps.

Table 7: Binary-feedback transcript-cap sweep for the linear softmax CLIP head on the expanded five-class benchmark. The cap is reset for each split; the actual reply counts are shown separately. Without abstention, ASR remains 1.0 and estimated gap upper bounds remain negative.

Cap q	**Train replies**	**Val. replies**	**Binary ASR**	$\widetilde{\mathrm{Gap}}$	ϵ_{95}
128	128	128	1.0000	−0.0249	0.6846
512	512	512	1.0000	−0.0253	0.7159
2,048	2,048	1,500	1.0000	−0.0263	0.7098
7,200	3,000	1,500	1.0000	−0.0258	0.7025

Finally, Table 8 reports a score-guided random-search sanity check. The search uses the deployed continuous target score to retain the best proposal and reports threshold-crossing success only; no $\widetilde{\mathrm{Gap}}$ was computed. It is not a purely binary private-oracle experiment and does not validate the hidden-state query bound in Theorem 8.

Table 8: Score-guided random-search sanity check. Entries are threshold-crossing success rates when continuous target scores guide proposal retention. This control reports neither a robust-gap estimate nor binary-only private-oracle performance.

Queries	Linear softmax	Two-layer MLP
128	0.3438	0.1562
512	0.4062	0.1875
2,048	0.4062	0.2500
7,200	0.4062	0.3125

D.3 Adversarially trained and stronger clean three-class models

The last appendix block addresses the clean-strength objection raised in Section 7.3. Table 9 gives the per-direction attack results for the adversarially trained three-class model. The table is needed because an aggregate ASR can hide whether one direction remains protected; here, every direction has negative estimated gap upper bounds under the evaluated budgets.

Table 9: Per-direction targeted attacks for the adversarially trained three-class model. This table supports Section 7.3: relative to the released checkpoint, clean performance improves substantially, but the model still fails under the same source–target attack directions.

Direction	Budget	ASR	$\widetilde{\mathrm{Gap}}$	Clean target acc.	Adv. target score
Human→AI	$2/255$	1.0000	-0.0576	0.6807	0.7383
Human→AI	$4/255$	1.0000	-0.0938	0.6807	0.7745
Human→AI	$8/255$	1.0000	-0.1236	0.6807	0.8042
AI→Human	$2/255$	0.9000	-0.0483	0.3730	0.4213
AI→Human	$4/255$	0.9817	-0.0809	0.3730	0.4539
AI→Human	$8/255$	1.0000	-0.1141	0.3730	0.4872
SD1.5→SDXL	$2/255$	1.0000	-0.0694	0.3899	0.4594
SD1.5→SDXL	$4/255$	1.0000	-0.1305	0.3899	0.5204
SD1.5→SDXL	$8/255$	1.0000	-0.1847	0.3899	0.5747
SDXL→SD1.5	$2/255$	0.9267	-0.0238	0.3623	0.3861
SDXL→SD1.5	$4/255$	0.9933	-0.0421	0.3623	0.4044
SDXL→SD1.5	$8/255$	1.0000	-0.0615	0.3623	0.4238

Across five seeds, full-batch retraining yields test accuracy 0.570 ± 0.047 and fake AUC 0.905 ± 0.021, whereas the stronger clean-only model reaches 0.870 ± 0.003 and 0.9823 ± 0.0002. The representative clean-only model used in Section 7.3 reaches clean accuracy 0.8689 and fake AUC 0.9822. Its targeted-attack evaluation still yields minimum ASR 0.9667 at $2/255$, ASR 1.0 for all four directions by $4/255$, and non-positive empirical gap upper bounds throughout. Together with Table 9, these controls support the main-text conclusion that clean-strength improvements alone do not remove the interface-learnability failure mode.

E Empirical Bracketing of Barrier I

This appendix connects the minimax statement in Section 4 to the empirical quantities reported in Section 7 and Appendix D. The attackability radius itself is difficult to estimate for realistic images. The following inequalities describe a conceptual bracket; obtaining tight computable bounds for realistic images remains unresolved. The image experiments instead estimate attack-derived upper bounds for fixed verifiers, subject to the finite-sample interpretation in Section 7. By Theorem 1, for any verifier family $\mathcal{F} \subseteq [0,1]^{\mathcal{X}}$,

$$\sup_{f \in \mathcal{F}} \mathrm{Gap}_{s,t}(f) \le \Delta_{s,t}.$$

Thus the true robust gap of any fixed deployed verifier, evaluated against the full admissible attack class, lower-bounds the ceiling. An evaluated heuristic attack only supplies an upper bound on the fixed-verifier gap through $\widetilde{\mathrm{Gap}}$; it does not thereby lower-bound $\Delta_{s,t}$.

On the attack side, if $\widetilde{\mathcal{Q}}_s \subseteq \mathcal{Q}_s$ is any explicit parametric family of attacked source distributions, then

$$\Delta_{s,t} = \inf_{Q \in \mathcal{Q}_s} \mathrm{TV}(P_t, Q) \leq \inf_{Q \in \widetilde{\mathcal{Q}}_s} \mathrm{TV}(P_t, Q).$$

This is why explicit editors can upper-bound the statistical ceiling when their induced distributions can be compared to the target. Our image experiments do not estimate these bounds tightly enough to recover $\Delta_{s,t}$. Appendix H.2 separately computes both minimax sides in a finite-state setting with known distributions.

F Extension Beyond Finite Image Spaces

Section 3 assumes a finite image space to keep the main proof elementary and to match quantized digital images. The attackability-radius law itself is not tied to finiteness: it extends to more general measurable spaces under compactness in total-variation norm. This section records that extension so that Barrier I is understood as a distributional statement rather than an artifact of finite notation.

Proposition 16 (Barrier I beyond finite image spaces)**.** *Let $(\mathcal{X}, \mathcal{F})$ be a measurable space and let $\mathcal{P}(\mathcal{X})$ denote its probability measures. Fix a target distribution $P_t \in \mathcal{P}(\mathcal{X})$ and let $\mathcal{Q}_s \subseteq \mathcal{P}(\mathcal{X})$ be nonempty, convex, and compact in total-variation norm. Then*

$$\sup_{f:\mathcal{X}\to[0,1] \text{ measurable}} \left(\mathbb{E}_{P_t}[f] - \sup_{Q \in \mathcal{Q}_s} \mathbb{E}_Q[f] \right) = \inf_{Q \in \mathcal{Q}_s} \mathrm{TV}(P_t, Q).$$

Proof. Let $\mathcal{B}_b(\mathcal{X})$ be the Banach space of bounded measurable real-valued functions on $\mathcal{X}$ with the sup norm, and let

$$\mathcal{F}_{[0,1]} := \{f \in \mathcal{B}_b(\mathcal{X}) : 0 \leq f \leq 1\}.$$

Define

$$\Lambda(Q, f) := \mathbb{E}_{P_t}[f] - \mathbb{E}_Q[f].$$

The set $\mathcal{F}_{[0,1]}$ is convex, and $\mathcal{Q}_s$ is convex and compact by assumption. For fixed Q, the map $f \mapsto \Lambda(Q, f)$ is affine and continuous in sup norm. For fixed f, the map $Q \mapsto \Lambda(Q, f)$ is affine and continuous in total-variation norm. Sion's minimax theorem therefore yields

$$\sup_{f \in \mathcal{F}_{[0,1]}} \inf_{Q \in \mathcal{Q}_s} \Lambda(Q, f) = \inf_{Q \in \mathcal{Q}_s} \sup_{f \in \mathcal{F}_{[0,1]}} \Lambda(Q, f).$$

For fixed Q, the inner supremum is the standard variational characterization of total variation on measurable spaces, so the result follows. □

G Reproducibility, Societal Impact, and AI-Assisted Writing

The public source repository is `https://github.com/kaikaiyao/pixels-alone-provenance`. It contains core experiment code and configurations, including implementations of the finite-state, RGB ResNet-18, and paired image-quality evaluations in Appendix H. Datasets, model weights, and saved experimental outputs are not bundled. Reproducing the image experiments requires the corresponding external data and model assets.

Compute resources. The original image experiments used NVIDIA A100 GPU workers. The main controlled studies use frozen CLIP features and small verifier heads rather than end-to-end training of large generative models. The exact-recovery, interface-sweep, binary-feedback, score-guided random-search, and retraining controls are dominated by feature extraction, small-head training, and attack/evaluation loops. The additional ResNet-18 victim is trained from RGB pixels, and the finite-state study solves linear programs. No reported experiment requires training a new image generator or a large vision-language backbone from scratch.

Experimental uncertainty. The image attack results are descriptive evaluations on held-out samples; their gap estimates are not accompanied by population confidence bounds. The original stochastic retraining controls report mean and standard deviation over five seeds. The finite-state plug-in study uses 20 repeats per setting and reports mean absolute error against known population values. The additional ResNet-18 evaluation uses one seed; its generality and training variability remain unresolved.

Existing assets and licenses. The experiments use existing assets credited in the paper and source bundle: MS COCO prompts/images (COCO annotations under CC-BY 4.0, with images subject to their original Flickr licenses and terms), OpenAI CLIP (MIT-licensed repository and public model-release terms), Stable Diffusion 1.5 (CreativeML OpenRAIL-M), Stable Diffusion 2.1, SDXL, and PixArt-family generators (CreativeML Open RAIL++-M/OpenRAIL++ model-card terms), and detector baselines from their cited authors. The source release contains code and configurations; datasets, image assets, model checkpoints, and saved experimental outputs are not redistributed. Users must obtain upstream assets from the original providers under their original terms.

New assets. The new assets are experiment source code and configuration files, including implementations of the additional evaluations in Appendix H. Repository documentation describes the code and required external inputs; no new dataset or model checkpoint is released.

Human subjects and IRB. The work does not involve crowdsourcing, participant studies, intervention with human subjects, or collection of private user data. No IRB review was sought because the experiments use public prompts/images, synthetic generations, and offline research implementations.

This paper studies attacks on passive provenance systems and therefore has dual-use implications. Targeted-forgery techniques could be adapted to weaken real provenance or moderation pipelines. The analysis is defensive: it clarifies that clean provenance accuracy can be a misleading robustness signal and that public verifier interfaces can become a dominant source of fragility. All experiments were conducted offline on controlled benchmarks and research implementations rather than against live moderation systems or third-party services.

During manuscript preparation, large language models (LLMs) and AI agents were used as assistive tools for literature research and scoping, drafting code scripts for empirical validations, checking mathematical statements, plotting experimental data, and refining and polishing the manuscript's language and structure. The author has manually reviewed all content prepared with assistance from LLMs and AI agents, including text, code, tables, and figures, for correctness and consistency and takes full responsibility for the manuscript's content and integrity.

H Additional Protocols and Evaluations

H.1 Surrogate training and attack protocol

The controlled benchmark contains aligned real, Stable Diffusion 1.5, and SDXL images for 1,200 MS COCO prompts, split by prompt into 600/300/300 training/validation/test prompts. The expanded benchmark adds Stable Diffusion 2.1 and PixArt under the same split protocol. Training and validation interface transcripts remain separated by prompt. For exact softmax-head recovery, the attacker queries full probability vectors on public CLIP features, converts the replies to log-odds, and solves an overdetermined least-squares system for the relative head parameters. For learned emulators, we use a two-layer model over public features with a 256-unit GELU hidden layer, trained for 30 full-batch epochs using AdamW with learning rate 10^{-3} and weight decay 10^{-4}. The loss follows the released response: mean-squared error for probability vectors or scalar scores, binary cross-entropy for binary responses, and cross-entropy for label-only or abstaining responses.

On the expanded benchmark, a transcript cap q applies separately to each split, yielding $\min(q, 3000)$ training replies and $\min(q, 1500)$ validation replies. Thus a cap of 7,200 yields 4,500 replies in total for this interface sweep; it is distinct from the 7,200 probes in the exact-recovery experiment. These controlled interface wrappers are API-style abstractions that isolate the effect of released feedback. They do not evaluate a live commercial service, whose preprocessing, model updates, account controls, and query limits may differ.

After fitting the emulator, the attacker differentiates through the public feature extractor and recovered or learned head to maximize the chosen target score. The pixel attacks use targeted projected gradient descent (PGD) under ℓ_∞ budgets $\delta \in \{2/255, 4/255, 8/255\}$, with 40 steps, step size $\delta/4$, and five restarts. The deployed verifier then scores each returned image for the reported ASR and empirical gap estimate. The validation-set 95th-percentile emulator error, ϵ_{95}, measures held-out approximation error; it does not bound the uniform error ε over the full attack region required by Theorem 4. The interface sweep also does not estimate the average surrogate optimization error $\bar{\alpha}$ separately, so its diagnostics cannot be substituted into the theorem's $2\varepsilon + \bar{\alpha}$ guarantee.

H.2 Finite-state validation of Barrier I

We evaluate the minimax identity on $\mathcal{X} = \{0,1\}^6$, where distributions and reachable edits can be represented explicitly. The study uses five fixed source–target distribution pairs and Hamming edit sets $\Gamma_r(x) = \{z : d_H(x,z) \le r\}$ for $r \in \{0,1,2,3\}$, giving 20 population settings. For each setting, an attack-side transport linear program and an independently formulated verifier-side linear program compute opposite sides of Theorem 1.

The two programs can be written for any finite source and target mass functions p_s and p_t as follows. Let π_{xz} be the probability mass transported from source state x to output state z, and let $q_z = \sum_x \pi_{xz}$. The attack-side program is

$$\begin{aligned} \min_{\pi,q,v} \quad & \frac{1}{2}\sum_{z\in\mathcal{X}} v_z \\ \text{subject to} \quad & \pi_{xz} \ge 0, \qquad \sum_{z\in\mathcal{X}} \pi_{xz} = p_s(x), \\ & \pi_{xz} = 0 \quad \text{if } z \notin \Gamma_r(x), \qquad q_z = \sum_{x\in\mathcal{X}} \pi_{xz}, \\ & v_z \ge p_t(z) - q_z, \qquad v_z \ge q_z - p_t(z). \end{aligned} \tag{1}$$

Its optimum is $\inf_{Q\in\mathcal{Q}_s} \mathrm{TV}(P_t, Q)$. The verifier-side program introduces u_x to upper-bound the largest admissible target score for source state x:

$$\begin{aligned} \max_{f,u} \quad & \sum_{z\in\mathcal{X}} p_t(z) f_z - \sum_{x\in\mathcal{X}} p_s(x) u_x \\ \text{subject to} \quad & 0 \le f_z \le 1, \qquad 0 \le u_x \le 1, \\ & u_x \ge f_z \quad \text{for every } z \in \Gamma_r(x). \end{aligned} \tag{2}$$

Its optimum is $\sup_f \mathrm{Gap}_{s,t}(f)$ because an optimal u_x can be chosen as the largest score among the admissible edits.

Across the 20 population settings, the maximum absolute discrepancy between the two independently computed values is 4.44×10^{-16}. We also replace the source and target distributions with empirical histograms and compare the resulting plug-in radius with the known population value, using 20 repeats per setting. Table 10 reports the mean absolute error across these settings and repeats. The decreasing error checks estimation in this finite domain; it does not estimate $\Delta_{s,t}$ for real high-dimensional image distributions.

Table 10: Finite-state plug-in estimation of the attackability radius. Mean absolute error is aggregated over 20 source–target/edit-radius settings and 20 repeats per setting. The population attack-side and verifier-side optima agree to a maximum absolute discrepancy of 4.44×10^{-16}.

Sample size n	**Mean absolute error**
250	0.02094
1,000	0.01008
5,000	0.00451

H.3 Transfer to a non-CLIP victim

To test whether transfer requires CLIP features in the victim, we evaluate an ImageNet-pretrained ResNet-18 [55] trained end-to-end on RGB pixels and fixed during attack evaluation. It reaches validation/test fake AUCs of 0.9830/0.9862 and test balanced accuracy of 93.58% at the validation-selected threshold of 0.8480. The victim returns one continuous fake-class score per query. The transcript contains 9,000 scalar replies: a clean image and one uniformly perturbed $\ell_\infty \leq 8/255$ variant for each of 3,000 training and 1,500 validation images. The attacker fits a two-layer emulator over frozen CLIP ViT-B/32 features and performs targeted PGD at $8/255$, with 40 steps and five restarts, without adaptive victim queries during attack optimization.

Table 11 separates overall ASR from success among sources classified correctly before attack. In the fake-to-real direction, 45 of 117 initially correct sources become accepted as real. In the reverse direction, none of the 122 initially correct sources succeeds; the sole overall success was already classified as the target before attack. The positive empirical gap estimates leave the sign of the true robust gap unresolved. These results support partial directional transfer across victim and emulator representations, under one victim architecture, one seed, the same-prompt benchmark, and the evaluated bounded pixel edits.

Table 11: Transfer from a CLIP-feature emulator to an RGB ResNet-18 victim at $\ell_\infty = 8/255$. Each direction contains 128 held-out sources. The initially-correct column excludes clean misclassifications and reports the number of induced target predictions. Positive empirical gap estimates do not certify positive robust separation.

Direction	**Overall ASR**	**Initially-correct ASR (count)**	$\widetilde{\mathbf{Gap}}$
Fake → Real	43.75%	38.46% (45/117)	$+0.5375$
Real → Fake	0.78%	0.00% (0/122)	$+0.9093$

H.4 Paired image quality and attack success

We measure structural similarity (SSIM) [57], learned perceptual image patch similarity with an AlexNet backbone (LPIPS-Alex) [56], and peak signal-to-noise ratio (PSNR) between clean and attacked images. Higher SSIM and PSNR, and lower LPIPS, indicate closer agreement under the respective paired metrics. This evaluation uses 16 held-out sources for each of the four attack directions in the controlled benchmark, at both $2/255$ and $8/255$. There are 64 source–target instances per budget and 128 attack instances in total. At each budget, 14 instances already have the target-side prediction before attack, leaving 50 initially correct instances for measuring induced success.

Table 12 reports induced success and ranges of per-direction median quality over all 16 instances in each direction. At $2/255$, the attack induces the target in 45 of 50 initially correct instances. Restricting the quality calculation to these 45 induced successes gives pooled medians of 0.9784 SSIM,

0.0063 LPIPS-Alex, and 43.82 dB PSNR. Increasing the budget to $8/255$ induces five additional successes, with lower SSIM and higher LPIPS medians. Thus the larger budget trades lower paired image quality for higher induced success. No evaluated instance exceeds its ℓ_∞ budget under the prespecified 10^{-6} tolerance. These metrics quantify fidelity to the source image; a claim about human perception or semantic preservation would require a separate evaluation.

Table 12: Paired image quality on the controlled benchmark. Each budget contains 64 attack instances, including 14 pre-existing target-side predictions. Induced ASR uses the remaining 50 initially correct instances. SSIM and LPIPS-Alex ranges are the minimum and maximum of the four direction-specific medians, each calculated over all 16 instances in that direction, rather than over successful attacks alone.

Budget	**Induced ASR (count)**	**Median SSIM range**	**Median LPIPS-Alex range**
$2/255$	90.0% (45/50)	0.9781–0.9849	0.0050–0.0081
$8/255$	100.0% (50/50)	0.8011–0.8534	0.0770–0.1365

Figures 4 and 5 show selected clean/adversarial pairs in all four directions using the same clean images at both budgets. The $2/255$ examples include an attack failure, while the selected $8/255$ examples induce the target prediction. Signed perturbations are displayed with the original eightfold amplification around mid-gray; these visualizations illustrate individual outcomes rather than estimate their prevalence.

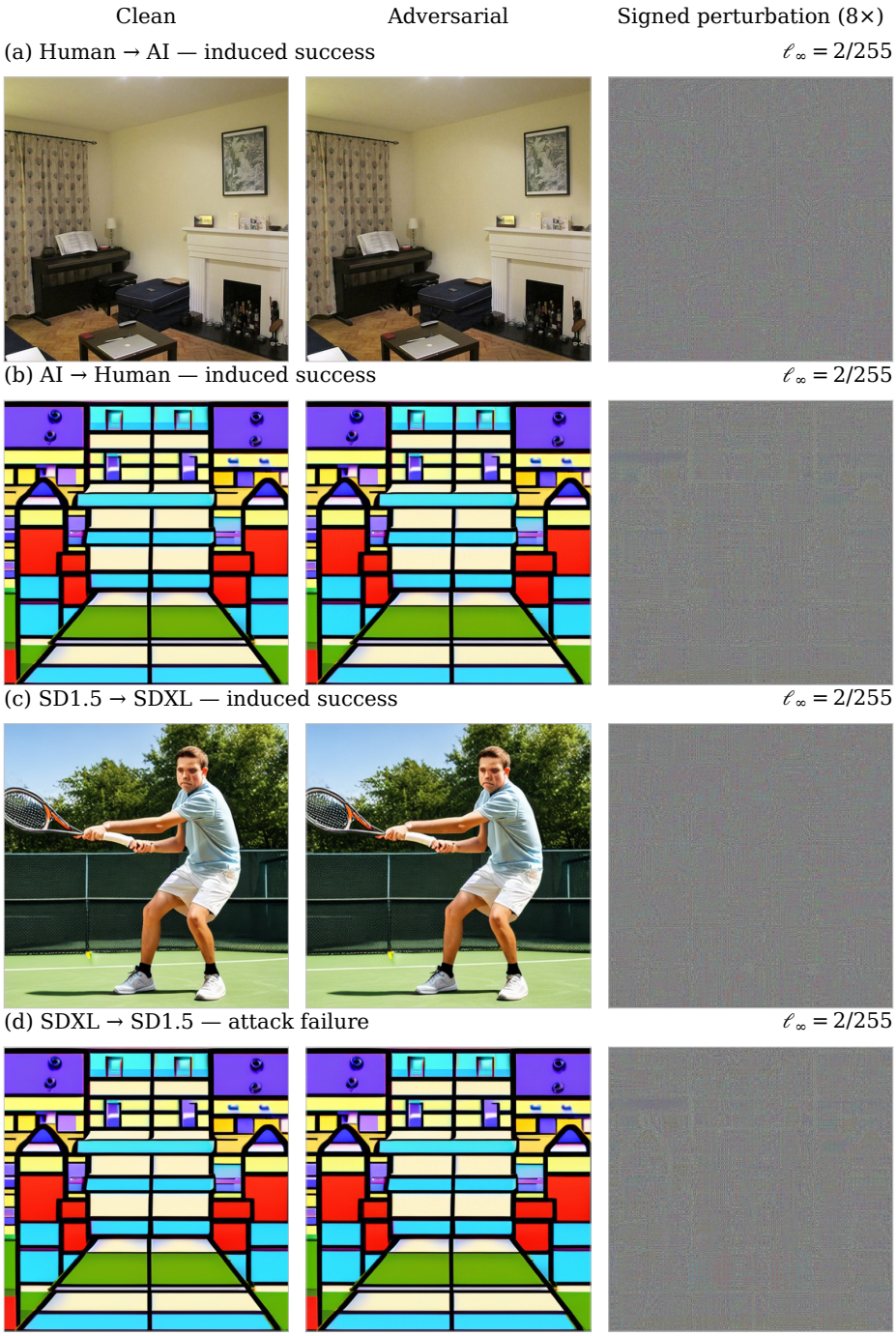


Figure 4: Selected paired examples in four attack directions at an ℓ_∞ budget of $2/255$. Columns show the clean image, the attacked image, and the signed perturbation amplified eightfold around mid-gray. The first three rows are induced successes and the SDXL→SD1.5 row is an attack failure. Figure 5 uses the same clean images at $8/255$.

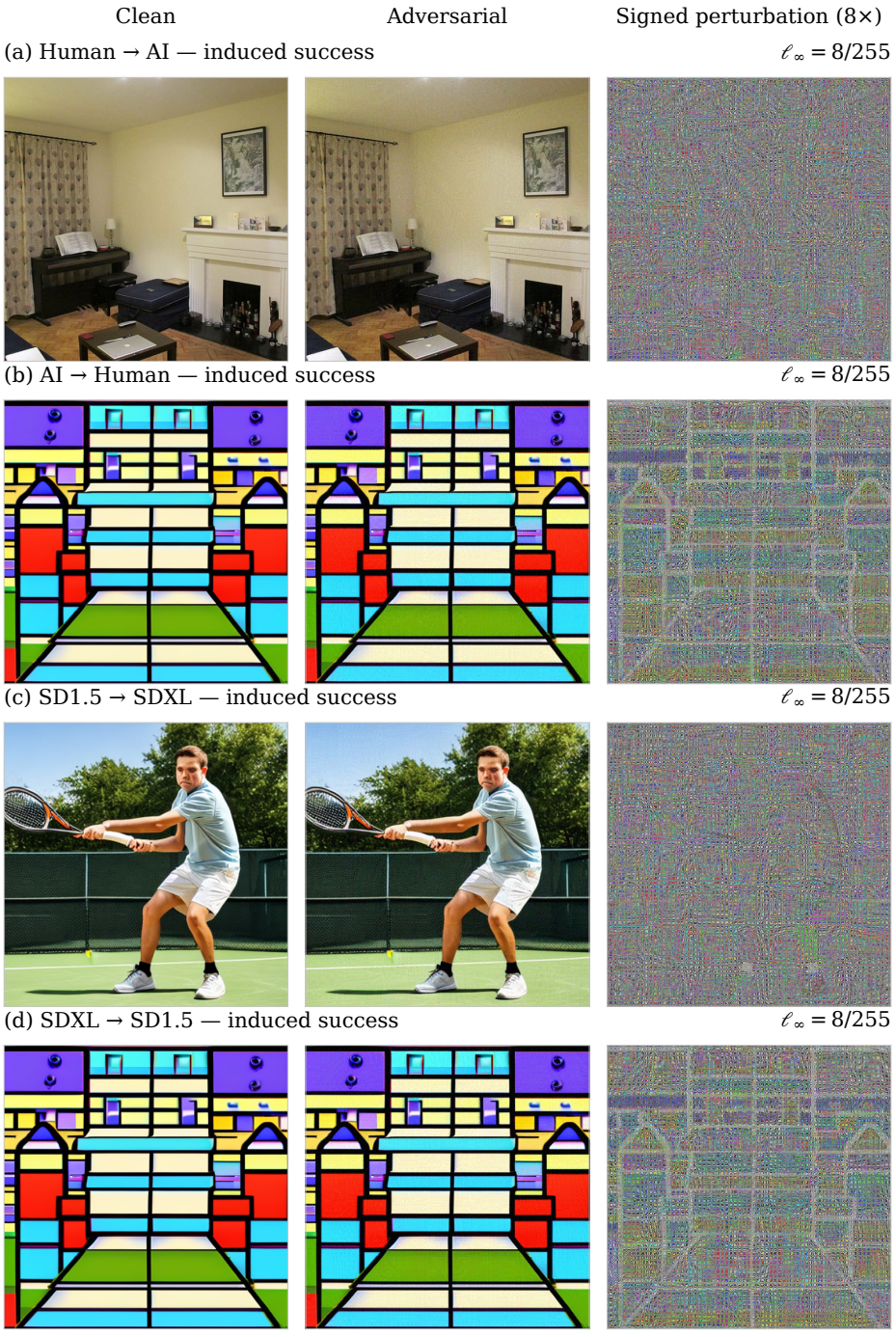


Figure 5: The same four clean images as in Figure 4, attacked at an ℓ_∞ budget of $8/255$. All four selected examples are induced successes. The paired views and amplified signed perturbations illustrate the larger changes at this budget; they do not establish human imperceptibility or semantic preservation.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction distinguish the exact source–target minimax characterization, the conditional emulator and private-interface results, and scoped empirical findings. Section 7 and Appendix H separate finite-state computation from fixed-verifier image experiments.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Section 8 distinguishes the finite-state calculation from the unestimated realistic-image radius, states the stylized private-interface assumptions, and limits the image evidence to the evaluated CLIP systems, one ResNet-18 victim and seed, bounded pixel edits, and local interface abstractions.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: The formal sections state the assumptions for the finite image space, attack family, emulator, score-recovery, and private-interface abstractions. Appendices A–F provide the complete proofs and the total-variation compactness extension.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Section 7 and Appendices D and H report benchmark splits, verifier families, interface-dependent surrogate losses, PGD settings, actual transcript counts, and the additional evaluation results. Appendix G links the public source code and describes the external data and model assets required for image experiments.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: We release experiment code and configurations at `https://github.com/kaikaiyao/pixels-alone-provenance`, including the additional finite-state, RGB ResNet-18, and paired image-quality evaluations. The release excludes datasets, image assets, model weights, and saved outputs, and is not a complete reproduction archive. Appendix G states this scope and the need for external data and model assets.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Section 7 reports the benchmark splits, interfaces, attack objective, PGD settings, and gap definition. Appendices D and H give the original per-direction tables, transcript counts, surrogate training settings, finite-state protocol, and additional ResNet-18 and paired-quality evaluations.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: Appendix D reports mean ± standard deviation over five retraining seeds, and Appendix H reports finite-state plug-in error over 20 repeats per setting. Appendix G states that the image attack estimates have no population confidence bounds and that the ResNet-18 evaluation uses one seed.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Appendix G identifies NVIDIA A100 workers for the original image experiments and describes feature extraction, small-head training, attacks, score-guided random search, and retraining. The additional finite-state linear programs and RGB ResNet-18 victim are identified separately.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Appendix G discusses dual-use risk, states that experiments were conducted offline on controlled benchmarks and research implementations, and states that no live moderation systems, third-party services, private user data, or human-subject experiments were used.

Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Appendix G discusses positive defensive uses for provenance evaluation and negative misuse risks from targeted-forgery techniques. It also frames the contribution as analysis of passive-provenance fragility rather than a deployment attack on live systems.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [Yes]

    Justification: Appendix G states that experiments are offline, controlled, and not run against live systems or third-party services. The public release is limited to experiment code, configurations, and documentation; it does not redistribute datasets, image assets, model weights, or saved experimental outputs.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.

- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: Appendix G lists the existing assets and licenses/terms for MS COCO, CLIP, Stable Diffusion-family/PixArt generators, and detector baselines, and states that full upstream datasets/checkpoints are not redistributed. The paper cites the corresponding original works where they are used.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: Appendix G describes the public experiment code and configurations, including implementations of the additional evaluations. Repository documentation describes the code, its scope, and required external inputs. No dataset or model checkpoint is released.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

Justification: Appendix G states that the work does not involve crowdsourcing, participant studies, intervention with human subjects, or collection of private user data. Therefore there are no participant instructions, screenshots, or compensation details to report.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: Appendix G states that the experiments use public prompts/images, synthetic generations, and offline research implementations, with no human-subject study or private user data. IRB approval is therefore not applicable to the described experiments.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: During manuscript preparation, large language models (LLMs) and AI agents were used as assistive tools for literature research and scoping, drafting code scripts for empirical validations, checking mathematical statements, plotting experimental data, and refining and polishing the manuscript's language and structure. The author has manually reviewed all content prepared with assistance from LLMs and AI agents, including text, code, tables, and figures, for correctness and consistency and takes full responsibility for the manuscript's content and integrity. See Appendix G.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.