NO SHORTCUTS TO TRUTH: EXPOSING AND ELIMINATING SHORTCUTS IN AI-GENERATED IMAGE DETECTION VIA CROSS-GENERATOR CONSISTENCY

Anonymous authors
Paper under double-blind review

ABSTRACT

Current synthetic image detectors frequently saturate on training generators but collapse on unseen ones. We trace this generalization failure to shortcut learning, a phenomenon where models capture spurious dataset correlations rather than genuine forgery evidence. We identify a fundamental structural asymmetry between these two feature types. Authenticity evidence is strictly confined to one side of the real and fake boundary, whereas shortcuts can manifest in both classes. Consequently, a shortcut-dependent detector yields fluctuating verdicts on the exact same real image when the paired synthetic generator changes. In contrast, an evidence-based model maintains prediction consistency. Building upon this insight, we propose Real Anchored Invariance, a novel data efficient multiple source framework that eliminates shortcuts without requiring prior knowledge of specific artifacts. Our method penalizes environment specific residuals along high variance directions of shared real features, dynamically upweights authentic instances that expose prediction disagreements, and simulates generator-level distribution shifts via episodic training. Extensive evaluations demonstrate that our approach achieves exceptional robustness and transferability across diverse unseen benchmarks and strictly held-out cutting-edge generators, even under extreme data scarcity.

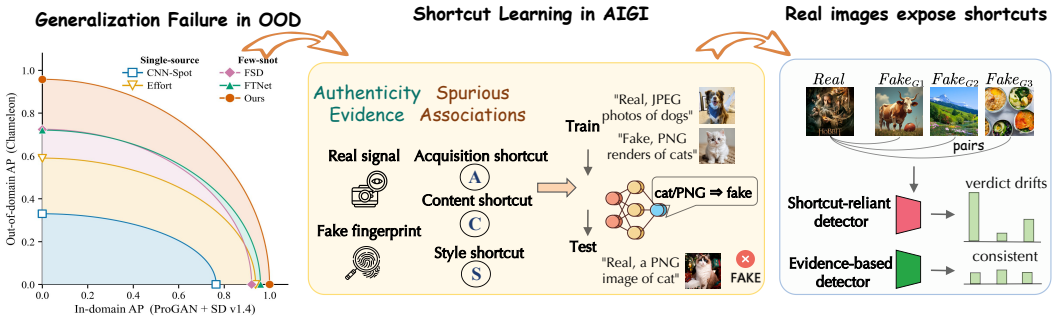

Figure 1: **Shortcut learning explains generalization failure in AIGI detection. Left**, in-domain and out-of-domain AP comparison. **Middle**, an example of shortcut learning, where all fake images are PNG and all real images are JPEG, causing the detector to mistake file format for authenticity. **Right**, a detector reliant on shortcuts drifts in its verdict on an unseen real image across varying pairings. Conversely, a detector capturing authenticity evidence maintains consistent predictions. Our method leverages this consistency across generators to expose shortcuts.

1 INTRODUCTION

Generative models now produce images that are often indistinguishable from real photographs (Rombach et al., 2022; Saharia et al., 2022; Karras et al., 2021), and their misuse for fraud and disinformation makes reliable detection of AI-generated images (AIGI) an urgent problem (Wang et al., 2025; Pei et al., 2026). The dominant approach trains a detector on a limited

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

set of generators, yet its performance often drops sharply when evaluated on unseen ones (Wang et al., 2019; 2023). This degradation can arise because the detector learns generator-specific cues that are predictive in the training environments but unstable across generators, thereby weakening generalization under generator shifts (Ojha et al., 2023; Wimmer et al., 2024).

We view this generalization failure through the lens of shortcut learning, which refers to reliance on spurious, environment-dependent cues rather than genuine authenticity evidence. As illustrated in Figure 1, a detector may rely on a file-format bias learned from the training data and consequently classify a real image as fake when that format appears at test time. This example shows that shortcut predictivity comes from dataset-specific co-occurrence. Authenticity evidence supports one side of the real-fake boundary, whereas shortcuts may occur in either class and lose their predictivity under distribution shift. To assess the effect, we categorize shortcuts into three types by their origin, namely Acquisition, Content, and Style, and conduct a preliminary study in Section 3. The results show that detectors readily rely on shortcuts, substantially degrading performance on unseen generators.

To reduce the impact of shortcut reliance, existing methods either predefine specific cues and design handcrafted features to suppress their influence (Zhong et al., 2023; Yan et al., 2025b), or preprocess and align real and fake data to reduce distributional biases (Chen et al., 2025b; Guillaro et al., 2025). However, both strategies require prior knowledge of the cues being targeted, whereas new generators continually introduce unseen shortcuts. This calls for a shortcut-agnostic training signal built on a stable reference. Such a reference is naturally provided by real images, which arise from physical imaging and belong to no generator, thereby offering generator-agnostic forensic cues. If detectors are independently trained by pairing the same real distribution with different generators, a detector that captures authenticity evidence should give consistent predictions for the same real image across pairings. In contrast, a detector relying on incidental cues will change its predictions as the paired generator and its associated evidence change.

Building on the above observation, we introduce **R**eal-**A**nchored **In**variance (RAIN), a shortcut-free framework that steers detection toward authenticity evidence for improved generalization. RAIN adopts few-shot multi-source training and requires only a few fake images from each generator. It employs a shared encoder and assigns a linear head to each generator environment. Specifically, RAIN consists of three components. *(i) Real-Anchored Alignment (RA)* penalizes differences among environment heads on the shared real-image distribution, encouraging predictions that remain stable across generator pairings. *(ii) Disagreement-Guided Reweighting (DGR)* identifies real images that induce substantial cross-head disagreement and strengthens the corrective supervision for the corresponding environment heads. *(iii) Leave-Environments-Out optimization (LEO)* repeatedly trains a temporary detector on a subset of generator environments and evaluates it on the held-out environments, guiding the encoder toward evidence that transfers across generator families.

To evaluate the effectiveness and robustness of our method, we conduct extensive experiments across multiple budgets. Across all budgets, RAIN achieves the highest average performance, and with only one fake sample per generator it attains an AP of **96.6%** on four out-of-distribution benchmarks. Our contributions are summarized as follows:

- We present the first systematic study of shortcut learning in AIGI detection. We construct a unifying acquisition, content, and style taxonomy, revealing how simplicity bias induces gradient starvation for genuine forensic cues.
- We propose RAIN, a novel shortcut-agnostic framework that turns cross-generator consistency on real images into a differentiable objective. By integrating distribution-level residual suppression, instance-level disagreement reweighting, and leave-environments-out optimization, RAIN effectively steers the shared representation toward universally transferable forgery evidence.
- Experiments show that RAIN outperforms both single-source detectors and few-shot detectors, with particularly pronounced gains on unseen generators and under real-world image degradations.

2 Related Work

Generalizable AIGI detection. Early detectors attribute cross-generator failure to insufficient feature transferability and pursue hand-designed cues such as frequency spectra (Qian et al., 2020; Frank et al., 2020), reconstruction residuals (Wang et al., 2023), and local texture statistics (Zhong

et al., 2023). Large pretrained encoders are subsequently adapted via low-rank, orthogonal, or null-space constraints to preserve transferable priors (Yan et al., 2025a; 2024b). At the coverage level, multi-generator benchmarks (Zhu et al., 2023; Chen et al., 2025a) and few-shot multi-source detectors (Wu et al., 2025; Lu et al., 2026) broaden the fake-side training distribution. However, these methods still pool all generators into a single empirical risk minimization (ERM), so any attribute that covaries with the label remains as fittable as genuine forgery evidence.

Shortcut mitigation. A parallel line of work attributes the failure to dataset-level spurious correlations. Grommelt et al. (2024) document systematic JPEG and resolution mismatches, and subsequent methods align the processing pipelines of real and fake images for each identified artifact (Chen et al., 2025b; Guillaro et al., 2025). Yet every such correction must first name the shortcut it targets, and new generators continually introduce unseen shortcuts that no fixed alignment can anticipate. Our work departs from both lines by exploiting a shortcut-agnostic training signal that requires no prior knowledge of any specific cue. Because real images arise from physical imaging and belong to no generator, they constitute a generator-agnostic reference.

3 Preliminaries

3.1 Problem Formulation

Let $\mathcal{X}$ denote the image space and $y \in \{0, 1\}$ the authenticity label, where $y = 0$ and $y = 1$ mark real and fake images, respectively. Real images are drawn from distribution P_R, while fake images are produced by a set of generators. We treat each training generator as an environment $e \in \mathcal{E}$ with its own fake distribution P_F^e. Under a few-shot multi-source protocol, each environment contributes exactly n fake images alongside an equal number of shared real images. Given a detector f, conventional training minimizes the empirical risk over the data, an objective that cannot separate authenticity evidence from any feature that happens to covary with the label. Our goal is instead to learn a detector f that remains reliable on generators absent from training, formalized as

$$\min_f \mathbb{E}_{x_R \sim P_R}\big[\ell(f(x_R), 0)\big] + \mathbb{E}_{x_F \sim P_F^{e'}}\big[\ell(f(x_F), 1)\big], \quad e' \notin \mathcal{E}, \tag{1}$$

where e' ranges over unseen generators and is never available during training.

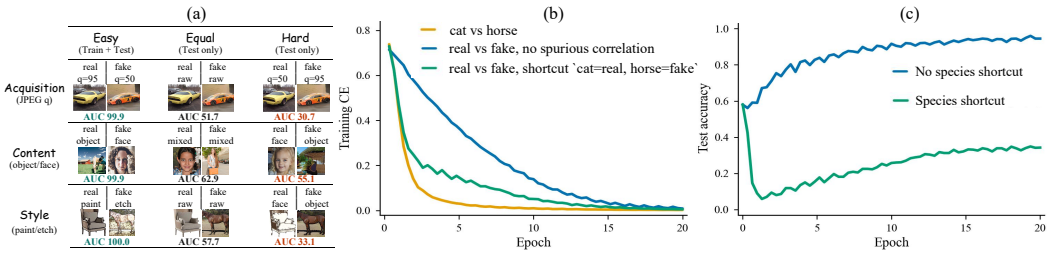

Figure 2: **Shortcuts dominate AIGI detection. (a)** For each ACS family, a representative shortcut is isolated by holding the other two families constant, and a ResNet-50 detector is trained and tested with the target shortcut preserved, removed, or reversed. **(b)** Training cross-entropy curves for a pure shortcut task, a real-versus-fake task with a species shortcut, and the same task without the shortcut. **(c)** Test accuracy on shortcut-free data tracked over training epochs for the shortcut-free and shortcut-contaminated detectors.

3.2 Shortcut Learning in AI-Generated Image Detection

Definition. In AIGI detection, shortcut learning refers to a detector relying on dataset-specific spurious correlations that happen to co-occur with the real/fake label. Such cues are predictive in distribution, yet they necessarily collapse under distribution shift. Authenticity evidence and shortcuts differ fundamentally in class support. Authenticity evidence is one-sided, exclusively in either real or fake images and remains anchored to one side of the real–fake boundary. Shortcuts are two-sided, since they can appear in either class. Under this criterion, generator fingerprints are

not shortcuts. Although often treated as generator-specific cues to be suppressed, they exist only in synthetic images and transfer well within a generator family.

The ACS Taxonomy. To characterize shortcuts systematically, we group them into three families. **A. Acquisition shortcuts** arise from how images are captured and stored, namely spatial resolution, JPEG quality factor, and file format. They are inherited whenever real and fake images pass through different pipelines. **C. Content shortcuts** stem from imbalances in what is depicted, namely subject category, scene composition, and spatial layout. They become label-aligned through skewed data content or prompt bias. **S. Style shortcuts** originate from how an image looks, namely color palette, illumination pattern, and rendering style. Generators and datasets favor these unevenly. Every family is equally realizable in a real or a fake image. Figure 2 (a) confirms that standard detectors readily absorb all three families. More details are provided in Appendix B.1.

Simplicity bias and gradient starvation. Shortcut features are not merely available but are in fact preferred. Because they are lower in complexity than authenticity cues, they reduce the loss faster and dominate the optimization trajectory. Figure 2 (b) makes this explicit. The loss curve of a shortcut-contaminated real-versus-fake task is nearly indistinguishable from that of the pure shortcut task, and both are fit within a handful of iterations, well before any authenticity signal has emerged. Once the shortcut has driven the loss close to zero, gradients for the remaining features vanish, so the detector never acquires them. Figure 2 (c) shows the consequence on shortcut-free test data. The shortcut-free detector improves monotonically, whereas the contaminated detector collapses in the first epoch and then merely overfits, never learning to discriminate real from fake.

Two requirements for a detector. The above observations reveal two core challenges. *First*, the detector must avoid shortcuts. A shortcut correlates with the label solely through dataset composition and remains specific to the chosen fake counterpart. Since real images are generator-agnostic and shared across all pairings, a robust detector must judge them identically regardless of the training counterpart. *Second*, avoiding shortcuts is necessary but insufficient, as the detector must also learn to generalize by transferring the acquired forgery evidence to unseen generator families.

4 METHODOLOGY

RAIN meets both challenges within a single framework, as illustrated in Figure 3. A shared encoder $\phi_\theta : \mathcal{X} \to \mathbb{R}^d$ maps an image to $z = \phi_\theta(x)$, each environment owns a linear head, and the deployed detector uses the environment-averaged weight:

$$h_e(z) = w_e^\top z + b_e, \qquad h_0(z) = \bar{w}^\top z + b_0, \qquad \bar{w} = \mathbb{E}_e[w_e], \tag{2}$$

with $s_e(x) = h_e(\phi_\theta(x))$ denoting the logit of environment e. All heads are trained on the same real images and differ only in their fake counterpart, so their disagreement on a real image is precisely the drift that we identify as a shortcut. Real-Anchored Alignment (RA) suppresses this drift at the level of the real-image distribution, Disagreement-Guided Reweighting (DGR) corrects it at the level of individual real images, and Leave-Environments-Out (LEO) drives ϕ_θ toward evidence that survives a change of generator. At inference, only ϕ_θ and h_0 are retained.

4.1 Real-Anchored Alignment

Since $\bar{w}$ is the average of the environment weights, each weight decomposes into a consensus direction and a zero-sum residual:

$$w_e = \bar{w} + \delta_e, \qquad \mathbb{E}_e[\delta_e] = 0. \tag{3}$$

The deployed head $h_0(z) = \bar{w}^\top z + b_0$ uses only the consensus $\bar{w}$. On a real image, the consensus term $\bar{w}^\top z$ is identical across heads, so any image-dependent disagreement is due to $\delta_e^\top z$. Because the environment bias b_e shifts all real images equally and cannot rank them, shortcuts that exploit variation inside P_R necessarily live in δ_e. RA therefore constrains these residuals directly. Let $z_R = \phi_\theta(x_R)$ and define the real-image statistics

$$\mu_R = \mathbb{E}_{x_R \sim P_R}[z_R], \qquad \tilde{z}_R = z_R - \mu_R, \qquad \Sigma_R = \mathbb{E}_{x_R \sim P_R}\left[\tilde{z}_R \tilde{z}_R^\top\right]. \tag{4}$$

Centering across environments makes the mean residual response vanish for any fixed feature. Centering the features themselves removes the uniform offset already absorbed by b_e and retains only

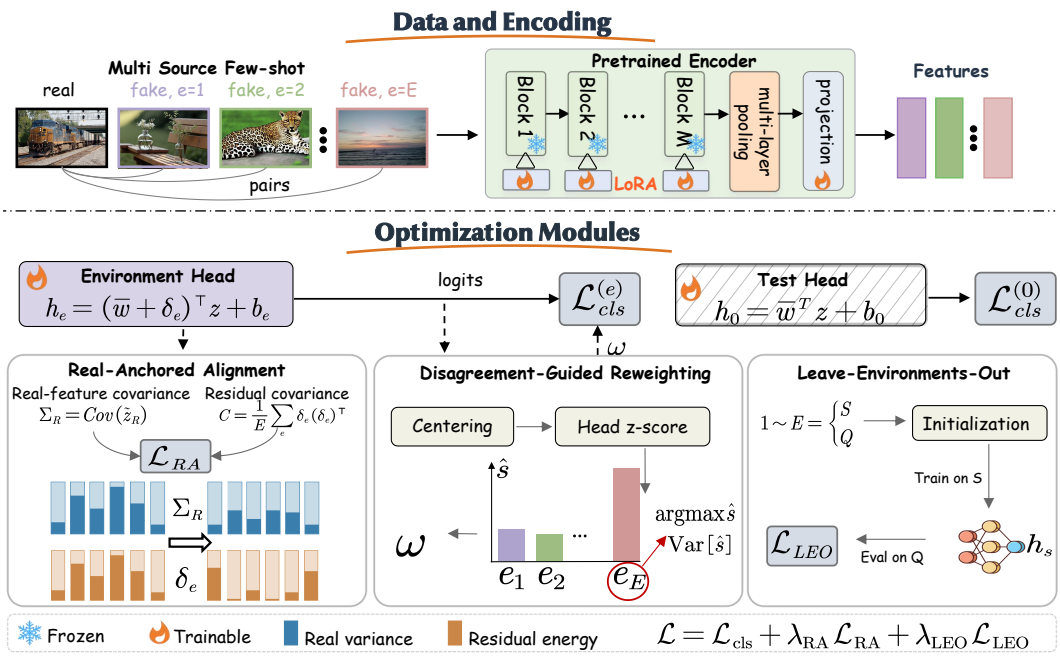

Figure 3: **Overview of RAIN. Top:** Multi-source few-shot samples are processed by a frozen backbone augmented with trainable LoRA modules to extract features. **Bottom:** Each environment has a dedicated linear head h_e, while a test head shares the mean weight $\bar{w}$ but uses a separate bias b_0. RA penalizes residual energy on directions where real images vary while leaving directions that are nearly constant on real images free to carry forgery cues. Blue bars show the variance of real features along each feature direction, and orange bars show the projection of δ_e onto the same directions. DGR upweights real instances where heads diverge most, and LEO episodically holds out training environments to steer the encoder toward transferable forgery evidence.

intra-real variation. The objective then reduces to a covariance form

$$\mathcal{L}_{\text{RA}} = \mathbb{E}_{x_R \sim P_R}\Big[\operatorname{Var}_e\left[\delta_e^\top \tilde{z}_R\right]\Big] = \mathbb{E}_{x_R}\mathbb{E}_e[(\delta_e^\top \tilde{z}_R)^2] = \mathbb{E}_e\left[\delta_e^\top \Sigma_R\, \delta_e\right]. \tag{5}$$

The covariance weighting is the key design choice. RA penalizes δ_e along directions where real features vary, so cues that fluctuate among real images are suppressed, while generator fingerprints that remain nearly constant on P_R stay admissible. Since $\sum_e \delta_e = 0$, the constraint does not move $\bar{w}$. Moreover, because Σ_R depends on ϕ_θ, RA also compresses real variation along the residual directions through the encoder. At equilibrium, the residual response $\delta_e^\top \tilde{z}_R$ approaches zero for almost every real image, yet δ_e itself need not vanish. Environment heads retain directions that are nearly constant on P_R but discriminative on their respective fake distributions. The complete derivation is provided in Appendix C.2.

4.2 Disagreement-Guided Reweighting

DGR corrects shortcut responses at the instance level. Since all heads share the same real images, substantial cross-head disagreement indicates that a specific head relies on an environment specific shortcut. DGR locates such images, identifies the head most inclined to classify them as fake, and strengthens the classification gradient of that head on these specific instances.

For a real image x_R, we standardize the logit of each head by subtracting the running mean of real scores and dividing by the batch standard deviation to normalize the scale. The resulting $\hat{s}_e(x_R)$ measures how unusually fake x_R appears to head e relative to its typical real response. The disagreement on x_R is the cross-head variance $d(x_R) = \operatorname{Var}_e[\hat{s}_e(x_R)]$, and the responsible head is $e^\star(x_R) = \arg\max_e \hat{s}_e(x_R)$. Head e assigns a weight to x_R proportional to

$$\omega_e(x_R) \;\propto\; 1 + \alpha\, d(x_R)\, \mathbb{1}[e = e^\star(x_R)]. \tag{6}$$

The weights are normalized to preserve the class balance, where α controls the reweighting strength. The class-balanced loss of environment e becomes

$$\mathcal{L}_{\text{cls}}^{(e)} = \tfrac{1}{2}\,\mathbb{E}_{x_F \sim P_F^e}\big[\ell(s_e(x_F), 1)\big] + \tfrac{1}{2}\,\mathbb{E}_{x_R \sim P_R}\big[\omega_e(x_R)\,\ell(s_e(x_R), 0)\big]. \tag{7}$$

DGR redirects the real image gradient of each head toward the instances that expose its shortcut while leaving the fake image term unchanged.

4.3 Leave-Environments-Out and Overall Objective

To address insufficient transfer to unseen generators, LEO simulates generator-level distribution shift during training. In each episode, the environment set $\mathcal{E}$ is randomly partitioned into a subset S and a subset Q with $S \cup Q = \mathcal{E}$ and $S \cap Q = \varnothing$. In the inner level, a temporary detector h_S is fitted from scratch on the features ϕ_θ of fake images from S and the shared real images, by solving a class-balanced ridge regression in closed form. In the outer level, h_S is evaluated on fake images from the held-out environments Q and the shared real images, and the resulting loss is back-propagated through the ridge solution to update ϕ_θ. This yields the bilevel objective

$$\mathcal{L}_{\text{LEO}} = \mathbb{E}_{(S,Q)}\Big[\ell\big(h_S \circ \phi_\theta;\; Q \cup P_R\big)\Big], \qquad h_S = \text{Ridge}\big(\phi_\theta(S \cup P_R)\big). \tag{8}$$

After each outer update, h_S is discarded and a new partition is sampled. Repeating this procedure over many partitions steers the encoder toward detection evidence that transfers across generator environments. The complete optimization procedure is provided in Appendix C.3.

During training, we optimize the overall objective

$$\mathcal{L} = \mathcal{L}_{\text{cls}} + \lambda_{\text{RA}}\mathcal{L}_{\text{RA}} + \lambda_{\text{LEO}}\mathcal{L}_{\text{LEO}}, \qquad \mathcal{L}_{\text{cls}} = \mathcal{L}_{\text{cls}}^{(0)} + \sum_{e \in \mathcal{E}} \mathcal{L}_{\text{cls}}^{(e)}. \tag{9}$$

Here $\mathcal{L}_{\text{cls}}^{(0)}$ is the loss of the test head, while $\mathcal{L}_{\text{cls}}^{(e)}$ denotes the loss of each environment head. DGR is applied exclusively to the real-image term of $\mathcal{L}_{\text{cls}}^{(e)}$, ensuring that the test head remains unaffected by the instance-level weights. At inference, we retain only the image encoder and the test head, then average the logits from five crops to obtain the final prediction.

5 Experimental Results

5.1 Settings

Baselines. We compare against nine off-the-shelf detectors spanning two training paradigms. Single-source detectors are trained on massive numbers of images paired with a single generator, including CNN-Spot (Wang et al., 2020), UnivFD (Ojha et al., 2023), NPR (Tan et al., 2024), AIDE (Yan et al., 2024a), SAFE (Li et al., 2025a), and Effort (Yan et al., 2024b). Multi-source few-shot detectors adapted from tiny support sets across multiple generators comprise FSD (Wu et al., 2025), FTNet (Yao et al., 2026), and Fleet (Wang et al., 2026). All baselines are evaluated using their official implementations, and the same few-shot support set is shared across all multi-source methods to ensure a fair comparison.

Datasets and protocol. We assess generalization along two dimensions. In the in-domain setting, single-source detectors use their official pretrained weights, few-shot detectors adapt to the AIGIBench (Li et al., 2025b) support set, and all methods are evaluated on the AIGIBench query set. In the out-of-domain setting, the same models are tested on four external benchmarks, namely GenImage (Zhu et al., 2023), Chameleon (Yan et al., 2024a), AIGCDetectBenchmark (Zhong et al., 2023), and CO-SPY (Cheng et al., 2025). To explore generalization to advanced unseen generators, we further curate a held-out set of 14 recent models from EvalGEN (Chen et al., 2025b) and AIGenImages2026 (Pantsios et al., 2026), none of which appear in any support set.

Metrics and implementation details. Following standard practice (Yan et al., 2024a), we report Accuracy and Average Precision. Unless otherwise specified, all reported metrics are averaged across the real and fake images to ensure a balanced evaluation. We adopt DINOv3 (Siméoni et al., 2025) as the frozen vision encoder and insert LoRA of rank 16 into all blocks. We pad input images to 224×224 and apply random cropping, random horizontal flipping, and random JPEG compression. The model is optimized using AdamW with a learning rate of 1×10^{-4}. All experiments are conducted on a single NVIDIA A6000 GPU, requiring approximately 1.2 hours for training.

Table 1: **In-domain detection AP (%) on AIGIBench spanning five families.** All few-shot baselines use 10 shots; our method is additionally evaluated at 1 and 5 shots. Best and second best results per column are in **bold** and underlined.

			GAN						Diffusion						
	Method	Venue	ProGAN	R3GAN	SG3	SG-XL	StyleSwin	WFIR	FLUX	MJ	GLIDE	DALLE3	Imagen3	SD3	SDXL
Single-source	CNN-Spot	CVPR'20	99.9	52.7	73.1	64.2	76.5	50.0	72.3	59.8	60.0	68.6	57.4	73.1	64.2
	UnivFD	CVPR'23	99.9	91.2	84.5	93.3	95.2	82.0	79.5	61.5	80.3	76.3	79.3	87.2	88.0
	NPR	CVPR'24	**100.0**	61.1	91.7	75.3	94.9	65.5	99.0	76.9	94.3	70.0	94.4	97.2	94.4
	AIDE	ICLR'25	99.6	97.1	91.4	93.2	89.3	90.8	93.4	83.0	97.7	63.1	95.2	98.3	95.7
	SAFE	KDD'25	**100.0**	98.0	97.5	93.8	99.1	73.8	99.7	96.0	97.9	48.2	98.2	98.8	99.9
	Effort	ICML'25	**100.0**	98.2	98.9	97.0	99.1	**100.0**	74.4	90.0	96.5	60.6	97.8	97.9	98.0
Few-shot	FSD-10	ICML'25	48.5	37.1	46.1	42.1	44.0	43.5	94.7	96.1	91.2	91.5	81.3	87.6	95.8
	FTNet-10	AAAI'26	97.6	77.8	87.2	84.4	89.0	64.8	99.0	92.9	97.4	83.9	91.8	97.1	99.6
	Fleet-10	ICML'26	99.6	86.9	98.1	85.2	83.7	81.3	**99.8**	**99.9**	97.5	**100.0**	**98.8**	**99.9**	**100.0**
	Ours-1	Ours	99.7	94.3	96.2	95.0	96.4	98.8	96.1	97.9	97.7	98.5	88.6	99.0	99.6
	Ours-5	Ours	99.9	99.9	99.7	99.7	**99.9**	**100.0**	99.0	99.1	99.5	99.8	94.7	**99.9**	**100.0**
	Ours-10	Ours	99.9	**100.0**	**99.8**	**99.8**	**99.9**	**100.0**	99.3	99.4	**99.6**	99.9	97.0	**99.9**	**100.0**

			Face Swapping					ID Personalization					In-the-wild		
	Method	Venue	E4S	FSwap	InSwap	SimSwap	BFace	Inf-ID	InstID	IP-Adp	PMaker	BLIP	SocRF	CommAI	**Mean**
Single-source	CNN-Spot	CVPR'20	68.9	58.7	77.9	70.0	73.4	49.5	80.2	65.8	58.2	92.9	50.6	59.1	67.1
	UnivFD	CVPR'23	57.1	52.4	40.2	40.4	35.3	89.6	93.5	87.3	72.3	95.8	55.2	73.2	75.6
	NPR	CVPR'24	34.4	43.6	40.7	42.7	34.7	80.4	79.2	91.7	43.6	**100.0**	68.4	62.9	73.9
	AIDE	ICLR'25	44.3	56.3	54.6	62.7	54.2	94.7	96.3	95.4	95.6	95.5	65.0	61.0	82.7
	SAFE	KDD'25	42.7	64.4	60.3	58.7	41.9	99.8	99.6	98.9	**99.8**	**100.0**	63.5	60.5	83.6
	Effort	ICML'25	76.9	85.8	83.1	89.7	66.1	85.7	87.1	98.1	69.2	**100.0**	65.2	65.2	87.2
Few-shot	FSD-10	ICML'25	85.3	92.8	94.0	90.0	91.7	97.8	91.1	95.0	66.3	60.7	65.0	75.7	76.2
	FTNet-10	AAAI'26	93.7	97.7	96.7	95.1	88.2	83.8	97.3	94.9	52.0	98.0	63.3	83.5	88.3
	Fleet-10	ICML'26	**99.8**	**99.8**	**99.7**	**99.8**	**99.8**	99.5	**99.8**	93.6	75.2	98.5	**99.0**	**97.6**	95.7
	Ours-1	—	96.5	91.0	90.9	92.0	94.5	98.8	97.4	97.4	89.2	94.8	92.9	90.9	95.4
	Ours-5	—	99.6	99.4	99.3	99.4	99.6	**100.0**	99.7	99.4	96.9	99.8	96.9	95.8	99.1
	Ours-10	—	99.7	99.7	99.6	99.7	**99.8**	**100.0**	**99.8**	**99.6**	98.0	99.9	98.0	96.4	**99.4**

5.2 Comparison with the State-of-the-Art

In-domain evaluation. Table 1 reports AP across 25 generators grouped into five families. Our method achieves the highest mean AP at every budget, surpassing the strongest few-shot competitor Fleet-10 by 3.7% despite using the same 10-shot budget, and exceeding the best single-source detector Effort by 12.2%. The results show that the performance of single-source detectors drops sharply on face-swapping and ID-personalization forgery types. On face-swapping, Effort peaks at 89.7%, whereas our 5-shot variant already reaches at least 96.9% on every entry. This discrepancy may stem from editor-based generators, whose artifacts differ substantially from text-to-image synthesis. The results confirm that aggregating cues from heterogeneous multi-source references is critical for powerful detectors. Additionally, our method exhibits strong data efficiency. Our 1-shot variant already outperforms all six single-source detectors and the 10-shot FSD, indicating that the learned multi-source representation transfers effectively from one sample per generator.

Out-of-domain evaluation. To evaluate detectors against real-world distribution shifts, we test on four benchmarks, as shown in Table 2. Our method attains the highest average AP at every shot budget and is the only detector that stays above 90% on all datasets. While single-source detectors suffer severe degradation on the challenging Chameleon, and the few-shot baseline FTNet struggles heavily on CO-SPY, our approach maintains exceptional stability. Table 3 reports 10-shot accuracy on 14 strictly held-out generators, including recent proprietary models like FLUX2-pro, GPT-Img1.5, and Gemini3. Existing baselines exhibit severe vulnerability to unseen architectures. FSD and FTNet collapse on the first ten models with high variance. Fleet improves the average to 91.2% but still drops below 88% on Gemini3 and SeeDream4.5. In contrast, our method consistently exceeds 94% across all 14 unseen generators. The results demonstrate that our method captures fundamental forgery cues rather than model-specific shortcuts, providing strong future-proofing against rapidly evolving generative technologies. Complete results are provided in Appendix E.

5.3 Robustness Evaluation

Real-world images are routinely re-compressed, re-encoded, and photometrically edited by social platforms, so we assess whether our detector remains reliable under such distortions. Following common practice, we consider two families of perturbations. Group A contains degradations,

Table 2: **Cross-dataset generalization on four out-of-domain benchmarks (AP%).** Left: single-source detectors trained on ProGAN / SD1.4. Right: few-shot detectors trained on AIGIBench with 1 / 5 / 10 shots. Best result per column in **bold**.

Method	GenImage	AIGCDB	Chameleon	CO-SPY
CNN-Spot	64.1	75.6	33.0	54.1
UnivFD	87.1	91.1	47.7	74.7
NPR	97.9	86.8	49.6	91.9
AIDE	97.4	**98.7**	**70.1**	96.8
SAFE	**99.2**	96.4	50.6	**99.6**
Effort	91.2	83.6	59.0	90.1

Method	GenImage			AIGCDB			Chameleon			CO-SPY		
	1	5	10	1	5	10	1	5	10	1	5	10
FSD	95.3	93.3	94.6	88.4	85.4	87.2	69.9	73.0	72.5	80.7	81.5	83.1
FTNet	87.1	90.1	91.2	84.5	89.9	92.0	76.7	74.9	72.0	52.0	76.4	82.7
Fleet	94.5	94.5	97.3	93.3	95.2	98.0	87.6	**95.3**	**95.9**	90.4	94.4	97.5
Ours	**97.7**	**98.7**	**97.8**	**98.2**	**99.0**	**98.8**	**90.5**	93.5	94.7	**99.9**	**100.0**	**100.0**

Table 3: **Cross-model generalization on 14 unseen generative models under the 10-shot setting (ACC%).** Best in **bold**, second best underlined.

Method	FLUX2-pro	GPT-Img1.5	Ideogram3	Gemini3	Firefly5	SeeDream4.5	Imagen4	Reve	Z-Image	HiDream	OmniGen	GoT	Infinity	NOVA	Avg
FSD	58.8	58.5	59.2	58.8	58.2	58.8	59.2	59.0	57.8	59.2	78.9	78.1	76.6	76.6	67.0±9.3
FTNet	43.1	43.1	43.2	42.9	41.8	43.2	43.2	43.2	42.7	43.1	77.5	77.5	77.2	77.3	55.4±16.4
Fleet	90.0	88.1	90.0	86.1	89.2	87.3	90.0	87.6	87.3	90.0	98.5	98.3	95.9	**98.4**	91.2±4.5
Ours	**95.3**	**95.4**	**98.0**	**95.4**	**97.2**	**95.8**	**95.8**	**95.1**	**94.6**	**96.1**	**99.6**	**99.3**	**99.4**	97.7	**96.1±1.9**

namely JPEG compression, Gaussian blur, Gaussian noise, and pixelation. Group B contains photometric transformations, namely contrast, saturation, and brightness. Each perturbation is applied at five increasing severity levels, with parameters detailed in Appendix E.3. Figure 4 reports AP under the four Group-A degradations. Single-source detectors collapse under distortions unseen during training, whereas our method degrades most gracefully across all four and stays above 80% even at the most aggressive level. These consistent margins suggest that our detector relies on degradation-invariant forensic cues rather than fragile high-frequency artifacts. Results on photometric perturbations are provided in Figure 8 of Appendix E.3.

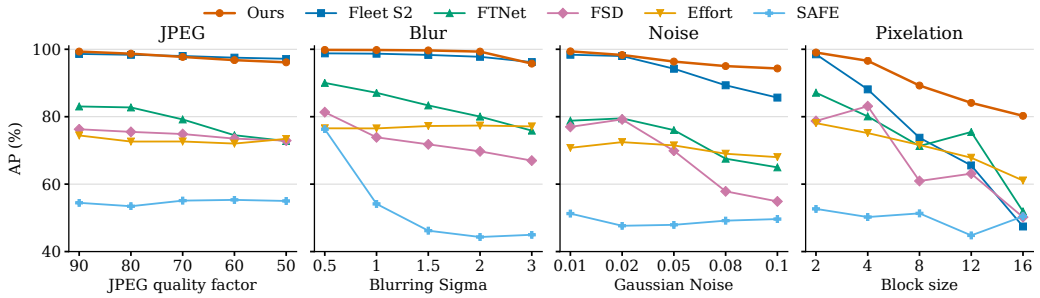

Figure 4: Robustness evaluation under four distinct image corruptions spanning five severity levels.

5.4 Ablation Studies

Hyperparameter analysis. In Figure 5 (a) and (b), the loss weights visibly affect the threshold-dependent FPR and FNR but leave the threshold-free AP nearly unchanged, since they shift score confidence relative to the fixed threshold without altering the real-fake ranking. For $\lambda_{\mathrm{RA}} \leq 1$, RA anchors real images far from the threshold by moving head residuals off the real-image variance directions compressed by LoRA, whereas overweighting RA shrinks these residuals and collapses the environment heads onto the mean head. At $\lambda_{\mathrm{LEO}} = 4$ the ridge transfer objective overrides the discrimination objective, leaving the test head to score fakes close to the decision threshold. Figure 5 (c) indicates that family diversity and the number of generators are both important, while added environments bring diminishing returns.

Component ablation. Table 4 shows that the few-shot baseline already surpasses the single-source baseline trained on abundant data, confirming the benefit of generator diversity. However, its high

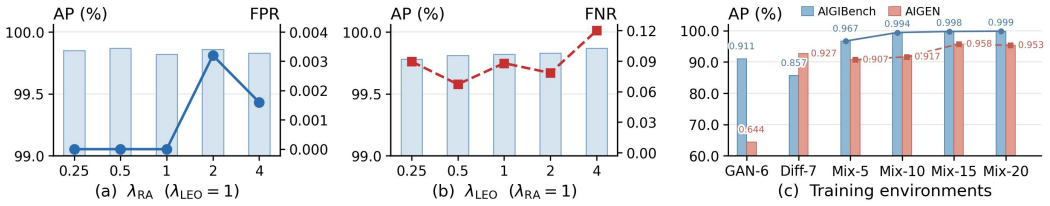

Figure 5: **Hyperparameter ablations of the 5-shot model.** (a) and (b) vary λ_{RA} and λ_{LEO} with the other fixed at 1. (c) varies environment diversity and count, and GAN-6 and Diff-7 use single-family generators.

Table 4: **Component ablation under the 5-shot setting.** Baseline 1 denotes a single-source detector trained on abundant data from one generator. Baseline 2 denotes naive multi-source few-shot training without the proposed components. Best in **bold**, second best underlined.

Setting			AP↑	FPR↓	FNR↓	ACC↑
Baseline 1			71.3	22.6	45.4	66.0
Baseline 2			83.7	4.1	58.6	68.6
RA	LEO	DGR				
✓	✗	✗	99.6	0.2	20.4	89.7
✗	✓	✗	99.8	0.7	12.7	93.3
✓	✓	✗	99.8	**0.0**	8.8	95.6
✓	✓	✓	**99.9**	**0.0**	**2.8**	**98.6**

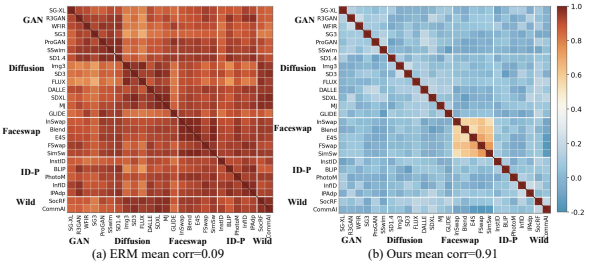

Figure 6: **Cross-probe ranking consistency on real images.** Each matrix entry denotes the Pearson correlation between standardized scores from two generator-specific linear probes evaluated on shared authentic inputs. High values indicate convergence toward a unified authenticity boundary rather than environment-dependent shortcut directions.

FNR indicates that naive multi-source training still generalizes poorly to unseen fakes. RA suppresses shortcut features by penalizing environment-specific head directions that capture real-image variation. LEO requires a probe fitted on seen generators to transfer to held-out generators, aligning representations with cross-generator authenticity cues and reducing missed fakes. DGR then reweights real images most often misjudged by the classifier heads, removing residual shortcut reliance and achieving the best overall performance.

Visualization. Figure 6 examines whether the learned representation still admits generator-dependent shortcuts on real images. Linear probes trained on disparate generators under standard ERM score identical real images nearly orthogonally, and cluster only around the face-swap family. Conversely, our framework elevates this inter-probe correlation to 0.91 across all families. By eliminating shortcut dimensions, the learned feature space ensures that classifiers trained against distinct generative distributions converge to a shared, generator-invariant authenticity boundary.

6 CONCLUSION

In this work, we first reframe the generalization bottleneck of AIGI detectors through the lens of shortcut learning driven by simplicity bias. We identify a structural asymmetry between one-sided authenticity evidence and two-sided spurious correlations, establishing a natural and shortcut-agnostic training signal. Because genuine photographs originate from physical processes independent of any synthesis algorithm, an evidence-grounded detector renders invariant verdicts on the real distribution regardless of the paired generative source. Translating this insight into an optimization objective, our framework RAIN eliminates shortcut reliance without requiring prior knowledge of specific artifacts. Extensive evaluations across challenging benchmarks and severe distribution shifts confirm that our method achieves superior cross-generator generalization under extreme data scarcity. Beyond immediate empirical gains, this work offers a broader theoretical lens for understanding distribution shifts in generative models. Future research will explore completely unsupervised shortcut decoupling to secure emerging multi-modal synthetic content such as videos and three-dimensional assets.

AI USE STATEMENT

Large language models were used to assist with the organization and refinement of the manuscript, including improving English expression, condensing overly verbose passages, checking notation and cross-section consistency, and suggesting LaTeX formulations and presentation improvements. They were also used to provide editorial feedback on the clarity of the argument and the presentation of figures and tables. The research questions, methodological design, implementation, experiments, data analysis, and scientific conclusions were developed and verified by the authors. All LLM-assisted content was reviewed and revised by the authors, who take full responsibility for the accuracy and integrity of the manuscript.

REPRODUCIBILITY STATEMENT

The problem setting and the few-shot multi-source protocol are formalized in Section 3.1, and all components of RAIN are described in Section 4, with complete derivations and the full training and inference procedures given in Appendix C. Datasets, baselines, backbone and adapter configuration, preprocessing, loss weights, optimizer settings, and random seeds are specified in Section 5.1 and Appendix D. Protocols of the shortcut probing experiments and the perturbation parameters of the robustness evaluation are detailed in Appendices B and E.3. The anonymous code repository is available at `https://anonymous.4open.science/r/RAIN-6CE1/`.

REFERENCES

Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019.

Ruoxin Chen, Jiahui Gao, Kaiqing Lin, Keyue Zhang, Yandan Zhao, Isabel Guan, Taiping Yao, and Shouhong Ding. AlignGemini: Generalizable ai-generated image detection through task-model alignment. *arXiv preprint arXiv:2512.06746*, 2025a.

Ruoxin Chen, Junwei Xi, Zhiyuan Yan, Ke-Yue Zhang, Shuang Wu, Jingyi Xie, Xu Chen, Lei Xu, Isabel Guan, Taiping Yao, et al. Dual data alignment makes ai-generated image detector easier generalizable. *arXiv preprint arXiv:2505.14359*, 2025b.

Siyuan Cheng, Lingjuan Lyu, Zhenting Wang, Xiangyu Zhang, and Vikash Sehwag. Co-spy: Combining semantic and pixel features to detect synthetic images by ai. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 13455–13465, 2025.

Joel Frank, Thorsten Eisenhofer, Lea Schönherr, Asja Fischer, Dorothea Kolossa, and Thorsten Holz. Leveraging frequency analysis for deep fake image recognition. In *International conference on machine learning*, pp. 3247–3258. PMLR, 2020.

Patrick Grommelt, Louis Weiss, Franz-Josef Pfreundt, and Janis Keuper. Fake or jpeg? revealing common biases in generated image detection datasets. In *European Conference on Computer Vision*, pp. 80–95. Springer, 2024.

Fabrizio Guillaro, Giada Zingarini, Ben Usman, Avneesh Sud, Davide Cozzolino, and Luisa Verdoliva. A bias-free training paradigm for more general ai-generated image detection. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18685–18694. IEEE, 2025.

Tero Karras, Miika Aittala, Samuli Laine, Erik Härkönen, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Alias-free generative adversarial networks. *Advances in neural information processing systems*, 34:852–863, 2021.

Jiaming Li, Hongtao Xie, Jiahong Li, Zhongyuan Wang, and Yongdong Zhang. Frequency-aware discriminative feature learning supervised by single-center loss for face forgery detection. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6454–6463. IEEE, 2021.

Ouxiang Li, Jiayin Cai, Yanbin Hao, Xiaolong Jiang, Yao Hu, and Fuli Feng. Improving synthetic image detection towards generalization: An image transformation perspective. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 1*, pp. 2405–2414, 2025a.

Ziqiang Li, Jiazhen Yan, Ziwen He, Kai Zeng, Weiwei Jiang, Lizhi Xiong, and Zhangjie Fu. Is artificial intelligence generated image detection a solved problem? *arXiv preprint arXiv:2505.12335*, 2025b.

Ming-Hui Liu, Harry Cheng, Xin Luo, Xin-Shun Xu, and Mohan S Kankanhalli. Towards generalizable deepfake detection via real distribution bias correction. *arXiv preprint arXiv:2603.14005*, 2026.

Jiajie Lu, Zhenkan Fu, Na Zhao, Long Xing, Xiangkun Wang, Kejiang Chen, Weiming Zhang, and Nenghai Yu. Liteupdate: A lightweight framework for updating ai-generated image detectors. *IEEE Transactions on Circuits and Systems for Video Technology*, 2026.

Utkarsh Ojha, Yuheng Li, and Yong Jae Lee. Towards universal fake image detectors that generalize across generative models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 24480–24489, 2023.

Thanasis Pantsios, Dimitrios Karageorgiou, Christos Koutlis, George Karantaidis, Olga Papadopoulou, and Symeon Papadopoulos. Automated in-the-wild data collection for continual ai generated image detection. In *Proceedings of the 5th ACM International Workshop on Multimedia AI against Disinformation (MAD '26')*, 2026.

Gan Pei, Jiangning Zhang, Menghan Hu, Zhenyu Zhang, Chengjie Wang, Yunsheng Wu, Guangtao Zhai, Jian Yang, and Dacheng Tao. Deepfake generation and detection: A benchmark and survey. *ACM Computing Surveys*, 58(11):1–41, 2026.

Jonas Peters, Peter Bühlmann, and Nicolai Meinshausen. Causal inference by using invariant prediction: identification and confidence intervals. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 78(5):947–1012, 2016.

Yuyang Qian, Guojun Yin, Lu Sheng, Zixuan Chen, and Jing Shao. Thinking in frequency: Face forgery detection by mining frequency-aware clues. In *European conference on computer vision*, pp. 86–103. Springer, 2020.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *2022 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 10674–10685. ieee, 2022.

Shiori Sagawa, Pang Wei Koh, Tatsunori B Hashimoto, and Percy Liang. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. *arXiv preprint arXiv:1911.08731*, 2019.

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35:36479–36494, 2022.

Chao Shuai, Shaojing Fan, Chenlin Zou, Bin Gong, Weichen Lian, Xiuli Bi, Zhenguang Liu, Zhongjie Ba, and Kui Ren. When detectors forget forensics: Blocking semantic shortcuts for generalizable ai-generated image detection. *arXiv preprint arXiv:2603.09242*, 2026.

Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, Francisco Massa, Daniel Haziza, Luca Wehrstedt, Jianyuan Wang, Timothée Darcet, Théo Moutakanni, Leonel Sentana, Claire Roberts, Andrea Vedaldi, Jamie Tolan, John Brandt, Camille Couprie, Julien Mairal, Hervé Jégou, Patrick Labatut, and Piotr Bojanowski. DINOv3, 2025. URL https://arxiv.org/abs/2508.10104.

Chuangchuang Tan, Yao Zhao, Shikui Wei, Guanghua Gu, Ping Liu, and Yunchao Wei. Rethinking the up-sampling operations in cnn-based generative network for generalizable deepfake detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 28130–28139, 2024.

Changshuo Wang, Jiangming Wang, Ke-Yue Zhang, Taiping Yao, Shouhong Ding, Ran Yi, and Lizhuang Ma. Breaking manifold continuity: Vector quantized modeling for real-centric deepfake detection. In *Forty-third International Conference on Machine Learning*.

Jiaan Wang, Sirui Liu, Yu Li, Kaiyuan Yang, Juan Cao, and Sheng Tang. Fleet: Few shots lead effective ai-generated image detection. *arXiv preprint arXiv:2606.31082*, 2026.

Junke Wang, Zhenxin Li, Chao Zhang, Jingjing Chen, Zuxuan Wu, Larry S Davis, and Yu-Gang Jiang. Fighting malicious media data: A survey on tampering detection and deepfake detection. *Proceedings of the IEEE*, 113(3):287–311, 2025.

Run Wang, Felix Juefei-Xu, Lei Ma, Xiaofei Xie, Yihao Huang, Jian Wang, and Yang Liu. Fakespotter: A simple yet robust baseline for spotting ai-synthesized fake faces. *arXiv preprint arXiv:1909.06122*, 2019.

Sheng-Yu Wang, Oliver Wang, Richard Zhang, Andrew Owens, and Alexei A Efros. Cnn-generated images are surprisingly easy to spot... for now. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8695–8704, 2020.

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

Georg Wimmer, Dominik Söllinger, and Andreas Uhl. The risk of image generator-specific traces in synthetic training data. *Proceedings Copyright*, 199:206, 2024.

Shiyu Wu, Jing Liu, Jing Li, and Yequan Wang. Few-shot learner generalizes across ai-generated image detection. *arXiv preprint arXiv:2501.08763*, 2025.

Jiazhen Yan, Ziqiang Li, Fan Wang, Boyu Wang, Ziwen He, and Zhangjie Fu. Dgs-net: Distillation-guided gradient surgery for clip fine-tuning in ai-generated image detection. *arXiv preprint arXiv:2511.13108*, 2025a.

Jiazhen Yan, Fan Wang, Weiwei Jiang, Ziqiang Li, and Zhangjie Fu. Ns-net: Decoupling clip semantic information through null-space for generalizable ai-generated image detection. *arXiv preprint arXiv:2508.01248*, 2025b.

Shilin Yan, Ouxiang Li, Jiayin Cai, Yanbin Hao, Xiaolong Jiang, Yao Hu, and Weidi Xie. A sanity check for ai-generated image detection. *arXiv preprint arXiv:2406.19435*, 2024a.

Zhiyuan Yan, Jiangming Wang, Zhendong Wang, Peng Jin, Ke-Yue Zhang, Shen Chen, Taiping Yao, Shouhong Ding, Baoyuan Wu, and Li Yuan. Effort: Efficient orthogonal modeling for generalizable ai-generated image detection. *arXiv preprint arXiv:2411.15633*, 2024b.

Shibo Yao, Renshuai Tao, Xiaolong Zheng, Chao Liang, and Chunjie Zhang. Leveraging failed samples: A few-shot and training-free framework for generalized deepfake detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 27764–27772, 2026.

Nan Zhong, Yiran Xu, Sheng Li, Zhenxing Qian, and Xinpeng Zhang. Patchcraft: Exploring texture patch for efficient ai-generated image detection. *arXiv preprint arXiv:2311.12397*, 2023.

Shuchang Zhou, Shangkun Wu, Jiwei Wei, Ke Liu, Ran Ran, Caiyan Qin, and Yang Yang. Frequency-aware semantic fusion with gated injection for ai-generated image detection. *arXiv preprint arXiv:2604.27875*, 2026.

Mingjian Zhu, Hanting Chen, Qiangyu Yan, Xudong Huang, Guanyu Lin, Wei Li, Zhijun Tu, Hailin Hu, Jie Hu, and Yunhe Wang. Genimage: A million-scale benchmark for detecting ai-generated image. *Advances in neural information processing systems*, 36:77771–77782, 2023.

A WHY PRIOR MITIGATION STRATEGIES FALL SHORT

A.1 LIMITATIONS OF AIGI-SPECIFIC METHODS

Existing strategies against shortcut learning have each improved generalization under the distribution shifts they target. They nevertheless share a common premise that the shortcut to be handled is specified before training, whether through a hand-designed feature, an added generator, or a data-alignment step. We examine this premise across three families and then state the observation that motivates our approach.

Representation-level methods change what the detector looks at. CNN-Spot (Wang et al., 2020) learns cues tied to GAN-family upsampling, which transfer within that family but less reliably beyond it. FGINet (Zhou et al., 2026) masks spectral bands to suppress specific bias, yet the masked range is set by the training generators, so frequency cues outside it remain available. GSD (Shuai et al., 2026) projects out a semantic direction from a frozen encoder, addressing content shortcuts while leaving acquisition and style cues unconstrained. Effort (Yan et al., 2024b) preserves a pretrained CLIP prior through orthogonal adaptation, which relies on that prior being largely free of label-correlated bias, a property pretraining does not guarantee.

Coverage-level methods change how much data the detector sees. GenImage (Zhu et al., 2023) broadens the fake side with multiple generators, and FSD (Wu et al., 2025), FTNet (Yao et al., 2026), and Fleet (Wang et al., 2026) adapt to a few samples from many generators, expecting new generators to be reached. However, an attribute shared by a subset of training generators but unrelated to authenticity remains fittable. Broader coverage reduces the number of untested shortcuts without changing the mechanism by which they are learned.

Bias-level methods change the data itself. Grommelt et al. (2024) document systematic JPEG-quality and resolution mismatches between real and fake images, and DDA (Chen et al., 2025b) and B-Free (Guillaro et al., 2025) align content, resolution, and compression through VAE reconstruction or self-conditioned generation. These works correctly identify a major source of shortcuts in the data construction pipeline, but the alignment pipeline is itself a generative process that may introduce artifacts of its own.

A.2 WHY GENERAL SHORTCUT-MITIGATION FRAMEWORKS DO NOT TRANSFER

Principled frameworks for spurious correlations, including Invariant Risk Minimization (Arjovsky et al., 2019), Distributionally Robust Optimization (Sagawa et al., 2019), and Invariant Causal Prediction (Peters et al., 2016), assume that environments are annotated for both classes and that the environment set is bounded. AIGI detection satisfies neither condition. Fake images can be partitioned by generator, but real images carry no generator identity, breaking the symmetric environment structure these objectives require. Moreover, fake generators evolve continuously, so invariance enforced over a fixed training set cannot anticipate novel forgery signatures from unseen architectures.

Because real images arise from physical imaging and belong to no generator (Frank et al., 2020), they provide a naturally generator-agnostic reference. Rather than enumerating fake environments or requiring real-side environment labels, our method enforces prediction consistency on shared real images across different paired generators, yielding a differentiable, shortcut-agnostic training signal without prior knowledge of specific artifacts.

B EMPIRICAL EVIDENCE FOR SHORTCUT LEARNING

B.1 VALIDATION OF THE ACS TAXONOMY AND SHORTCUT DOMINANCE

We verify that standard detectors absorb ACS-family shortcuts whenever they are present, and that this pattern generalizes across independent operators. Each ACS family is instantiated with one representative attribute and evaluated under a unified three-way protocol. *Easy*, shortcut shared between train and test; *Equal*, shortcut removed; *Hard*, shortcut reversed at test time. All experiments use a fully fine-tuned ImageNet-pretrained ResNet-50 with AdamW, trained for 8 epochs on 2,000 images per class and evaluated on 800 per class per split, reporting AUC as the threshold-free metric. Table 5 summarizes the per-family configuration.

Table 5: Shortcut instantiation per ACS family under the three-way protocol.

Family	**Easy** (shared)	**Equal** (removed)	**Hard** (reversed)
Acquisition (JPEG quality)	Real q=95; Fake q=50	Raw for both	Real q=50; Fake q=95
Content (subject category)	Real: object; Fake: face	Mixed subjects	Real: face; Fake: object
Style (rendering medium)	Real: painterly; Fake: sketch	No stylization	Real: sketch; Fake: painterly

To confirm that the reversal pattern is not an artifact of the first-round operators, we repeat the identical protocol with a second, disjoint operator per family, drawn from ForenSynths at native 256 resolution with no image overlap. Table 6 reports AUC across both rounds.

Table 6: Two-round shortcut injection across six disjoint operators. All six saturate under Easy and reverse under Hard.

Round	**Family**	**Operator**	**Easy**	**Equal**	**Hard**
1	Acquisition	JPEG quality (q=95 vs. q=50)	0.999	0.517	0.307
1	Content	Subject slot (face vs. object)	0.999	0.629	0.551
1	Style	Rendering medium (paint vs. etch)	1.000	0.577	0.331
2	Acquisition	File format (.png vs. .jpg)	1.000	0.720	0.015
2	Content	Category (animal vs. artifact)	1.000	0.612	0.023
2	Style	Palette (warm-earth vs. cool-cyan)	1.000	0.506	0.000

All six operators saturate under Easy and reverse under Hard, confirming that a standard detector fits these spurious correlations. The Equal column deviates from 0.5 in some cases because natural data cannot be perfectly stripped of residual shortcuts.

The experiments above already employ images drawn directly from AIGIBench and ForenSynths, confirming that detectors readily absorb whichever spurious correlation is present in the data. To further verify that such label-correlated shortcuts are not confined to our experimental constructions but already embedded in the standard benchmarks used by the community, we audit three widely adopted real-fake pairings.

- **Acquisition** (GenImage, SD1.4): Real images inherit JPEG from ImageNet; fakes are released as lossless PNG. File format alone is a trivial label-correlated cue.
- **Content** (AIGIBench, InstantID): Fakes are close-up facial portraits; paired reals mix street scenes, animals, and objects. Subject category is entangled with the label.
- **Style** (AIGIBench, DALL-E 3): Reals are natural photographs; fakes skew toward oil paintings and high-saturation illustrations. Rendering style acts as a spurious cue.

B.2 Universality of Simplicity Bias in Optimization Dynamics

Having established that ACS shortcuts dominate detector decisions, we now show that this dominance stems from a fundamental optimization phenomenon, namely simplicity bias. We conduct two probing experiments under an identical protocol. A single pool of 4,000 ProGAN images from AIGIBench is fixed, along with the same ImageNet-pretrained ResNet-18 initialization, the same data order, and the same SGD configuration with a 50-step linear warm-up over 20 epochs. The only variable is the training label. In the first experiment, we contrast a pure content task, cat versus horse, against a real-versus-fake task in which the four cells formed by crossing the real/fake label with the cat/horse label are equally populated. Test accuracy is tracked on the two label-conflicting cells, namely real images of horses and fake images of cats, to assess whether the authenticity signal can ever be recovered once dominated by the shortcut.

Figure 2 (b) reveals that the loss curve of the shortcut-contaminated real-versus-fake task is nearly indistinguishable from that of the pure shortcut task, and both converge within a handful of iterations, well before any authenticity signal has emerged. Once the shortcut has driven the loss close to zero, gradients for the remaining features vanish. Figure 2 (c) shows the consequence on shortcut-free test data. The shortcut-free detector improves monotonically, whereas the contaminated detector

collapses in the first epoch and then merely overfits, never learning to discriminate real from fake. To confirm that this simplicity bias is not an artifact of the specific content shortcut, we repeat

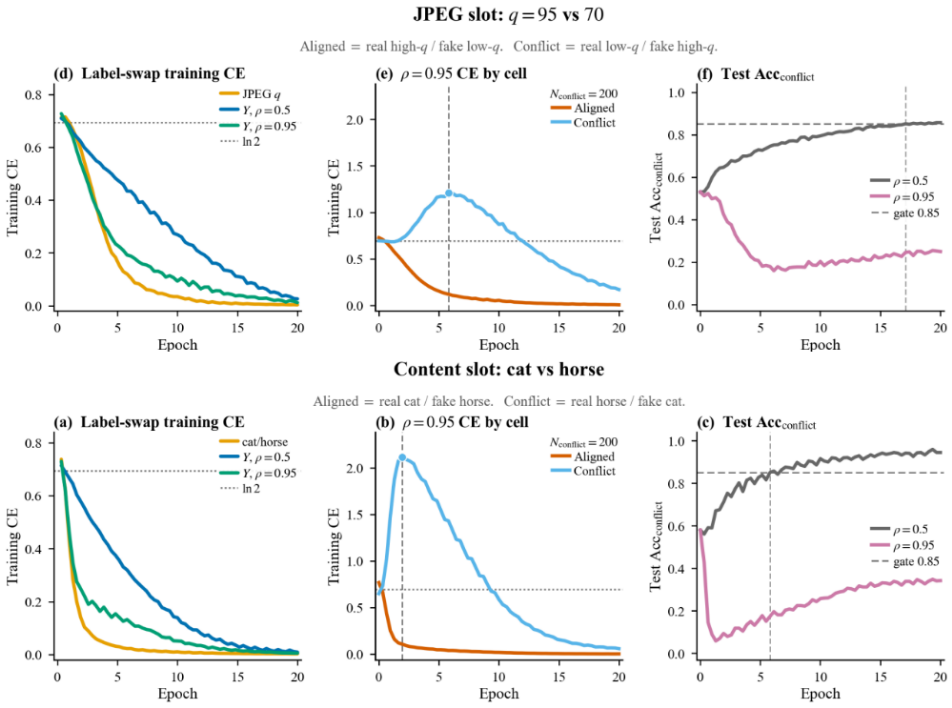

Figure 7: **Simplicity bias under an acquisition shortcut based on JPEG quality.** The loss dynamics and accuracy collapse mirror those observed with the content shortcut in Figure 2 (b), (c), confirming that simplicity bias is not confined to semantic features.

the identical protocol using an acquisition-type shortcut, namely JPEG quality with q=95 for one class and q=70 for the other. As illustrated in Figure 7, the JPEG shortcut reproduces the exact same qualitative signature. The pure shortcut task converges markedly faster than the balanced authenticity task. Under a skewed composition with ρ=0.95, aligned cells are fit immediately while conflicting cells exhibit a pronounced loss spike before being memorized. On the held-out test set, conflicting-cell accuracy drops within the first few epochs and never recovers.

Together, the content and acquisition experiments demonstrate that simplicity bias universally governs how standard detectors absorb structurally unrelated spurious correlations, motivating the shortcut-resistant training objectives introduced in Section 4.

C Theoretical Derivations

C.1 Notation

Table 7 summarizes the core notation used throughout the paper.

C.2 Real-Anchored Alignment

For each environment $e \in \mathcal{E}$, the prediction head is

$$h_e(z) = w_e^\top z + b_e. \tag{10}$$

The deployed head uses the mean environment weight and an independent bias:

$$\bar{w} = \mathbb{E}_e[w_e], \qquad h_0(z) = \bar{w}^\top z + b_0. \tag{11}$$

Define the environment-specific residual by

$$\delta_e = w_e - \bar{w}. \tag{12}$$

Table 7: **Core notation used in the paper.**

Symbol	Definition
$\mathcal{X}$	Image space.
x, y	Input image and authenticity label, where $y = 0$ denotes real and $y = 1$ denotes fake.
P_R	Distribution of real images shared across environments.
P_F^e	Distribution of fake images generated by environment e.
$\mathcal{E}$, e	Set of training environments and an individual environment.
e'	An unseen test environment, with $e' \notin \mathcal{E}$.
f	Generic detector used in the problem formulation.
ϕ_θ	Shared image encoder parameterized by θ.
$z = \phi_\theta(x)$	Image representation produced by the shared encoder.
d	Dimension of the image representation, with $z \in \mathbb{R}^d$.
$h_e(z)$, w_e, b_e	Linear prediction head for environment e and its parameters: $h_e(z) = w_e^\top z + b_e$.
$\bar{w}$, δ_e	Mean environment-head weight and environment-specific residual: $\bar{w} = \mathbb{E}_e[w_e]$ and $\delta_e = w_e - \bar{w}$.
$h_0(z)$, b_0	Deployed test head and its bias: $h_0(z) = \bar{w}^\top z + b_0$.
$d(x_R)$	Cross-head disagreement score for real image x_R.
$\omega_e(x_R)$	Normalized DGR weight assigned to real image x_R for environment head e.
S, Q	Support and query environment sets in a LEO episode.

By construction,

$$\mathbb{E}_e[\delta_e] = \mathbb{E}_e[w_e] - \bar{w} = 0. \tag{13}$$

For a real image $x_R \sim P_R$, let

$$z_R = \phi_\theta(x_R), \qquad \mu_R = \mathbb{E}_{x_R \sim P_R}[z_R], \qquad \widetilde{z}_R = z_R - \mu_R, \tag{14}$$

and define the covariance of centered real representations as

$$\Sigma_R = \mathbb{E}_{x_R \sim P_R}\left[\widetilde{z}_R \widetilde{z}_R^\top\right]. \tag{15}$$

The feature-dependent environment residual on a real image is

$$r_e(x_R) = \delta_e^\top \widetilde{z}_R. \tag{16}$$

For every fixed real representation, its environment mean is zero:

$$\mathbb{E}_e[r_e(x_R)] = \mathbb{E}_e[\delta_e]^\top \widetilde{z}_R = 0. \tag{17}$$

RA penalizes the variance of the residual response across environment heads:

$$\mathcal{L}_{\mathrm{RA}} = \mathbb{E}_{x_R \sim P_R}\left[\mathrm{Var}_e[r_e(x_R)]\right]. \tag{18}$$

Because the environment mean of $r_e(x_R)$ is zero,

$$\begin{aligned} \mathcal{L}_{\mathrm{RA}} &= \mathbb{E}_{x_R \sim P_R} \mathbb{E}_e\left[(\delta_e^\top \widetilde{z}_R)^2\right] \\ &= \mathbb{E}_e \mathbb{E}_{x_R \sim P_R}\left[\delta_e^\top \widetilde{z}_R \widetilde{z}_R^\top \delta_e\right] \\ &= \mathbb{E}_e\left[\delta_e^\top \Sigma_R \delta_e\right]. \end{aligned} \tag{19}$$

Since Σ_R is positive semidefinite, write

$$\Sigma_R = U \operatorname{diag}(\lambda_1, \ldots, \lambda_d) U^\top, \qquad \lambda_k \geq 0. \tag{20}$$

Then

$$\delta_e^\top \Sigma_R \delta_e = \sum_{k=1}^{d} \lambda_k \left(u_k^\top \delta_e\right)^2, \tag{21}$$

where u_k is the k-th eigenvector of Σ_R. Thus, residual components aligned with high-variance directions of the real-feature distribution receive larger penalties. For a fixed covariance matrix, the gradient with respect to the residual is

$$\nabla_{\delta_e} \left(\delta_e^\top \Sigma_R \delta_e\right) = 2\Sigma_R \delta_e. \tag{22}$$

Because the real representations and Σ_R depend on ϕ_θ, the RA term also contributes gradients to the shared encoder.

C.3 Leave-Environments-Out Optimization

At the beginning of each LEO episode, the training environments are randomly partitioned into disjoint support and query sets S and Q:

$$S \cup Q = \mathcal{E}, \qquad S \cap Q = \emptyset. \tag{23}$$

The support set contains fake samples from environments in S and shared real samples from P_R. Let the resulting class-balanced support set contain N_S samples (x_i, y_i), and define the augmented representation

$$\widetilde{z}_i = \begin{bmatrix} \phi_\theta(x_i) \\ 1 \end{bmatrix} \in \mathbb{R}^{d+1}. \tag{24}$$

Let $Z_S \in \mathbb{R}^{N_S \times (d+1)}$ contain the row vectors $\widetilde{z}_i^\top$, and let $y_S \in \mathbb{R}^{N_S}$ contain the labels. Class balance is represented by a diagonal matrix W_S whose entries are $1/(2N_R)$ for real support samples and $1/(2N_F)$ for fake support samples, where N_R and N_F are the numbers of real and fake support samples.

The temporary detector is a linear ridge predictor on the augmented features. To keep the bias unregularized, define

$$R = \begin{bmatrix} I_d & 0 \\ 0 & 0 \end{bmatrix}. \tag{25}$$

For $\lambda_{\text{ridge}} > 0$, the class-balanced support objective is

$$\min_{\psi} \; \frac{1}{2} \left\| W_S^{1/2} (Z_S \psi - y_S) \right\|_2^2 \frac{\lambda_{\text{ridge}}}{2} \psi^\top R \psi. \tag{26}$$

Its closed-form solution is

$$\psi_S^\star = \left(Z_S^\top W_S Z_S + \lambda_{\text{ridge}} R\right)^{-1} Z_S^\top W_S y_S. \tag{27}$$

Writing $\psi_S^\star = [w_S^{\star\top}, b_S^\star]^\top$, the temporary detector is

$$h_S(z) = w_S^{\star\top} z + b_S^\star. \tag{28}$$

The detector fitted on S is evaluated on fake samples from the held-out environments in Q and shared real samples from P_R. The LEO objective is

$$\mathcal{L}_{\text{LEO}} = \mathbb{E}_{(S,Q)} \left[\ell \left(h_S \circ \phi_\theta;\, Q \cup P_R\right)\right]. \tag{29}$$

The query loss differentiates through the closed-form support solution. In particular, for one support-query episode,

$$\nabla_\theta \ell = \frac{\partial \ell}{\partial \theta} + \left(\frac{\partial \psi_S^\star}{\partial \theta}\right)^\top \nabla_{\psi_S^\star} \ell. \tag{30}$$

The temporary detector is discarded after the outer update, while the updated encoder parameters are retained for the next episode.

C.4 Disagreement-Guided Reweighting

For a real image x_R, the logit of environment head e is

$$s_e(x_R) = h_e\big(\phi_\theta(x_R)\big). \tag{31}$$

Following the main text, the logit is standardized using the running mean of real scores and the mini-batch standard deviation:

$$\widehat{s}_e(x_R) = \frac{s_e(x_R) - \mu_e^{\mathrm{run}}}{\sigma_e^{\mathcal{B}} + \varepsilon}, \tag{32}$$

where $\varepsilon > 0$ avoids numerical instability. The disagreement score and the responsible environment head are

$$d(x_R) = \mathrm{Var}_e[\widehat{s}_e(x_R)], \qquad e^\star(x_R) = \arg\max_{e \in \mathcal{E}} \widehat{s}_e(x_R). \tag{33}$$

The unnormalized DGR weight for environment head e is

$$\widetilde{\omega}_e(x_R) = 1 + \alpha d(x_R)\mathbf{1}[e = e^\star(x_R)], \tag{34}$$

where $\alpha \geq 0$ controls the reweighting strength. Normalization over the real distribution preserves the class balance:

$$\omega_e(x_R) = \frac{\widetilde{\omega}_e(x_R)}{\mathbb{E}_{x_R \sim P_R}[\widetilde{\omega}_e(x_R)]}. \tag{35}$$

The resulting class-balanced loss for environment head e is

$$\mathcal{L}_{\mathrm{cls}}^{(e)} = \frac{1}{2}\mathbb{E}_{x_F \sim P_F^e}\left[\ell(s_e(x_F), 1)\right] + \frac{1}{2}\mathbb{E}_{x_R \sim P_R}\left[\omega_e(x_R)\ell(s_e(x_R), 0)\right]. \tag{36}$$

Thus, DGR changes the relative contribution of real-image examples for each environment head while leaving the fake-image term unchanged. The test-head classification term does not use these instance-level weights.

C.5 Training Procedure and Gradient Flow

At each training iteration, fake mini-batches are sampled from the training environments and one shared real mini-batch is sampled from P_R. The encoder maps all samples to representations $z = \phi_\theta(x)$. The test head and the environment heads then produce their classification logits.

For the shared real batch, RA computes the centered real representations and the environment-specific residual responses. DGR standardizes the real logits, computes $d(x_R)$ and $e^\star(x_R)$, and constructs the normalized weights $\omega_e(x_R)$. These weights are used only in the real-image terms of the environment-head classification losses.

Independently, a LEO episode partitions $\mathcal{E}$ into S and Q. A temporary detector is fitted on S by the closed-form ridge solution above, evaluated on $Q \cup P_R$, and differentiated through its support solution.

The total training objective is

$$\mathcal{L} = \mathcal{L}_{\mathrm{cls}} + \lambda_{\mathrm{RA}}\mathcal{L}_{\mathrm{RA}} + \lambda_{\mathrm{LEO}}\mathcal{L}_{\mathrm{LEO}}, \tag{37}$$

where

$$\mathcal{L}_{\mathrm{cls}} = \mathcal{L}_{\mathrm{cls}}^{(0)} + \sum_{e \in \mathcal{E}} \mathcal{L}_{\mathrm{cls}}^{(e)}. \tag{38}$$

The classification terms update the shared encoder and their corresponding heads. RA supplies gradients through the real-feature covariance and the environment-head residuals. LEO supplies gradients to the shared encoder through the closed-form support detector. After the update, the temporary LEO detector is discarded.

C.6 Optimization and Inference Settings

Training uses the objective defined above with the shared encoder, one head per training environment, and the deployed test head. The numerical optimizer, learning rate, batch size, loss coefficients, and other experimental settings are specified in the experimental setup.

At inference, all environment-specific heads and temporary LEO detectors are removed. The retained model consists only of ϕ_θ and h_0. Given an input image x, let $c_i(x)$ denote the five fixed crops used by the evaluation protocol. The final logit is the average crop logit:

$$s(x) = \frac{1}{5}\sum_{i=1}^{5} h_0\big(\phi_\theta(c_i(x))\big). \tag{39}$$

The fake-image probability is

$$p_{\text{fake}}(x) = \sigma\big(s(x)\big). \tag{40}$$

D Experimental Setup

D.1 Datasets

Following Fleet (Wang et al., 2026), we adopt AIGIBench to provide the multi-environment few-shot support set. To comprehensively evaluate generalization, we select four out-of-domain benchmarks that cover a wide spectrum of generative architectures, content domains, and distribution shifts, ensuring the validity of our conclusions.

- **Chameleon** (Yan et al., 2024a): Focuses on open-world, highly photorealistic AI-generated images. Both real and fake samples closely resemble in-the-wild web photographs rather than laboratory-synthesized images. The presence of heavy compression, mixed sources, and weak artifacts makes it a severe stress test that exposes shortcut failures, as many detectors saturating on AIGIBench suffer significant performance drops here.
- **GenImage** (Zhu et al., 2023): Utilizes real images from ImageNet as the natural counterpart and covers eight distinct generator families, including BigGAN, ADM, GLIDE, Stable Diffusion (SD), VQDM, and Midjourney. It evaluates whether the detector's ranking capability remains stable when shifting away from the object classification domain.
- **AIGCDetectBenchmark** (Zhong et al., 2023): Integrates early GANs (e.g., CycleGAN, StarGAN, StyleGAN) with modern diffusion models and datasets like WhichFaceIsReal into a unified directory. It tests whether the method merely memorizes new diffusion-specific traces or remains effective against older synthesis protocols.
- **CO-SPY** (Cheng et al., 2025): Aggregates recent open-source pipelines such as FLUX, Playground, PixArt, SegMoE, and SD3. The generator vendors and versions are deliberately disjoint from the training directory, specifically measuring the transferability across models that belong to the diffusion family but utilize entirely different weights and architectures.
- **EvalGEN** (Chen et al., 2025b): Includes updated models such as Flux, OmniGen, Infinity, and NOVA. These images are saved with JPEG compression, imposing both architectural updates and lossy acquisition distortions to prevent inflated performance observed only on pristine PNG laboratory images.
- **AIGenImages2026** (Pantsios et al., 2026): Employs commercial and closed-source generators from 2026, including Firefly, GPT Image, Gemini, and FLUX.2. These models are completely unseen during training, directly evaluating the detector's extrapolation capability to the latest production-level generative technologies.

D.2 Baseline Methods

We compare RAIN against nine representative detectors, all of which are classic or state-of-the-art methods spanning two training paradigms.

- **CNN-Spot** (Wang et al., 2020) is a classic baseline that trains a standard CNN classifier on ProGAN outputs, relying on GAN-family upsampling artifacts.
- **UnivFD** (Ojha et al., 2023) leverages a frozen CLIP feature space with a lightweight classifier to build a more universal fake image detector.
- **NPR** (Tan et al., 2024) rethinks the up-sampling operations in CNN-based generators and exploits neighboring pixel relationships as a generalizable forgery cue.
- **AIDE** (Yan et al., 2024a) combines low-level artifacts with high-level semantic features to improve detection across diverse generators.
- **SAFE** (Li et al., 2025a) improves generalization from an image transformation perspective, aligning preprocessing to suppress dataset-specific biases.
- **Effort** (Yan et al., 2024b) is a recent SOTA method that preserves a pretrained CLIP prior through efficient orthogonal adaptation for generalizable detection.
- **FSD** (Wu et al., 2025) is a few-shot learner that generalizes across AI-generated image detection by adapting to a handful of samples from many generators.
- **FTNet** (Yao et al., 2026) is a few-shot and training-free framework that leverages failed samples for generalized deepfake detection.
- **Fleet** (Wang et al., 2026) is the current SOTA few-shot detector that uses only a few shots to achieve effective AI-generated image detection.

D.3 Implementation Settings and Details

Data sampling. We follow the few-shot multi-source protocol established by prior methods. Training samples are drawn from a subset of AIGIBench categories, and the training and test splits are strictly disjoint. Under the n-shot setting, each generator environment contributes exactly n fake images together with n shared real images, so that every mini-batch contains an equal number of real and fake samples. All sampling is governed by a fixed random seed to ensure that the set of fake images seen during training remains identical across runs.

Data preprocessing. No spatial resizing is applied at any stage. Images whose shorter side is below 224 pixels are zero-padded; all others are used at their native resolution. During training we apply random cropping, random horizontal flipping, and random JPEG compression with probability $p = 0.5$ and quality factor $q \sim \mathcal{U}[60, 100]$. Pixel values are normalized with the ImageNet channel-wise mean and standard deviation. At inference, five crops are averaged to produce the final logit.

Backbone and adapter. The visual encoder is DINOv3 ViT-L/16 with all pretrained weights frozen. LoRA modules are injected into every one of the 24 transformer blocks, targeting the query, key, value, output projections and both feed-forward matrices. The LoRA rank is $r = 16$ with scaling factor $\alpha = 32$ and dropout rate 0.1. Intermediate features from blocks $\{6, 12, 18, 24\}$ are aggregated and projected into a 512-dimensional embedding that serves as the shared representation z.

Optimization. The model is trained with AdamW under a cosine learning-rate schedule with a peak learning rate of 1×10^{-4}. The global training seed is 0, governing weight initialization, data augmentation order, and optimizer stochasticity. The loss weights are set to $\lambda_{\mathrm{RA}} = \lambda_{\mathrm{LEO}} = 1$. The LEO support–query partition seed is 7, so that the episodic environment split is reproducible. All experiments are conducted on a single NVIDIA A6000 GPU and require approximately 1.2 hours of training.

E Additional Experimental Results

E.1 Complete In-Domain Results

Table 8 complements Table 1 in the main paper by reporting Accuracy (Acc) on the same 25 AIGIBench generators. RAIN achieves the highest mean accuracy at every evaluated budget. In particular, RAIN with one sample per generator reaches 89.4% mean Acc, exceeding Fleet-10 by 1.1 points despite using one tenth of its fake training samples. Increasing the budget to 5 and 10 shots further improves the mean Acc to 95.6% and 96.1%, respectively. Together with the AP results

in the main paper, these results show that RAIN improves both ranking quality and thresholded classification performance across heterogeneous generator families.

Table 8: **In-domain detection Accuracy (%) on AIGIBench spanning five families.** All few-shot baselines use 10 shots; RAIN is additionally evaluated at 1 and 5 shots. Best and second best results per column are in **bold** and underlined.

			GAN						Diffusion						
	Method	Venue	ProGAN	R3GAN	SG3	SG-XL	StyleSwin	WFIR	FLUX	MJ	GLIDE	DALLE3	Imagen3	SD3	SDXL
Single-source	CNN-Spot	CVPR'20	97.6	50.4	55.8	52.8	52.6	49.8	57.4	52.3	51.1	53.9	51.2	55.8	52.8
	UnivFD	CVPR'23	98.4	83.5	79.6	84.6	86.4	70.0	80.0	65.3	76.7	75.1	78.9	84.5	84.7
	NPR	CVPR'24	99.4	50.8	78.4	60.3	85.7	51.6	95.2	68.8	82.5	57.1	85.9	91.9	86.6
	AIDE	ICLR'25	97.2	92.9	88.1	88.7	83.7	71.4	88.0	76.4	93.4	55.1	89.8	94.3	93.5
	SAFE	KDD'25	99.8	92.2	93.0	90.1	95.4	54.1	95.4	84.7	91.7	46.1	**92.9**	93.7	96.3
	Effort	ICML'25	**100.0**	95.3	95.5	89.6	96.8	95.5	61.4	80.3	89.1	54.8	90.7	92.1	91.8
Few-shot	FSD-10	ICML'25	74.4	56.5	73.4	76.7	78.2	51.1	87.1	76.9	88.9	85.8	84.8	87.1	87.2
	FTNet-10	AAAI'26	68.7	80.0	82.1	79.1	80.5	64.8	81.6	66.6	83.5	75.6	80.9	80.8	80.9
	Fleet-10	ICML'26	93.3	68.9	88.1	63.9	69.6	61.2	**97.8**	**98.0**	88.8	**98.8**	92.8	**98.4**	**98.9**
	Ours-1	Ours	95.5	88.5	90.0	88.8	91.6	93.3	91.7	92.4	93.6	96.0	78.3	96.0	97.0
	Ours-5	Ours	96.8	**97.8**	98.3	**97.8**	**98.1**	**98.7**	95.2	95.0	96.6	97.7	83.5	97.9	98.2
	Ours-10	Ours	96.1	97.4	**98.6**	97.6	97.6	98.4	96.0	95.9	**96.9**	97.7	89.4	97.8	98.1

			Face Swapping					ID Personalization					In-the-wild		
	Method	Venue	E4S	FSwap	InSwap	SimSwap	BFace	Inf-ID	InstID	IP-Adp	PMaker	BLIP	SocRF	CommAI	**Mean**
Single-source	CNN-Spot	CVPR'20	51.1	50.3	54.5	52.1	52.4	49.7	53.2	52.0	50.1	77.2	51.1	51.3	55.1
	UnivFD	CVPR'23	57.0	53.1	43.7	43.7	35.0	84.5	85.4	82.6	69.3	88.6	54.4	67.0	72.5
	NPR	CVPR'24	45.0	48.1	47.8	47.4	44.5	63.9	63.8	82.4	48.1	99.2	59.1	54.0	67.9
	AIDE	ICLR'25	44.3	52.1	50.9	54.9	51.5	92.2	91.8	90.0	91.7	96.4	57.8	54.1	77.6
	SAFE	KDD'25	43.8	49.0	48.4	48.0	43.8	95.6	95.7	94.2	**95.3**	**99.7**	58.2	54.5	78.1
	Effort	ICML'25	56.4	56.1	58.0	66.5	52.7	69.8	71.2	92.5	59.8	**99.7**	58.1	52.6	77.1
Few-shot	FSD-10	ICML'25	37.4	66.1	46.3	52.2	38.8	85.2	86.9	84.0	55.0	93.4	68.0	66.9	71.5
	FTNet-10	AAAI'26	75.6	85.0	84.3	84.4	74.0	80.5	82.1	81.5	54.1	90.8	54.2	63.2	76.6
	Fleet-10	ICML'26	**98.2**	**98.3**	**98.0**	**98.2**	**98.6**	96.2	97.6	73.3	54.3	93.4	**91.8**	**91.6**	88.3
	Ours-1	—	90.8	82.7	81.8	83.1	84.0	96.3	95.4	94.1	80.3	90.3	82.4	80.2	89.4
	Ours-5	—	97.4	97.0	96.6	96.4	96.9	**98.2**	97.6	**96.5**	88.7	97.6	88.4	87.7	95.6
	Ours-10	—	97.3	97.3	97.3	97.0	97.3	97.8	**97.7**	**96.5**	92.1	97.7	87.5	89.9	**96.1**

E.2 Complete Out-of-Domain Results

Table 9 reports the Accuracy corresponding to the AP results in Table 2. RAIN obtains the highest accuracy in 10 of the 12 few-shot dataset–budget comparisons. Its 5-shot variant achieves the strongest average accuracy across the four benchmarks (94.0%), followed by RAIN-10 (93.5%). On Chameleon, RAIN leads at 1 shot and maintains substantially stronger accuracy on CO-SPY. These results indicate that the cross-generator consistency learned by RAIN transfers to thresholded decisions under diverse dataset shifts.

Table 9: **Cross-dataset generalization Accuracy (%) on four out-of-domain benchmarks.** Left: single-source detectors trained on ProGAN or SD1.4. Right: few-shot detectors trained on AIGIBench with 1, 5, or 10 samples per generator. Best and second best few-shot results per column are in **bold** and underlined.

Method	GenImage	AIGCDB	Chameleon	CO-SPY
CNN-Spot	55.6	61.3	56.9	73.1
UnivFD	66.2	77.3	57.4	80.1
NPR	88.6	82.7	59.8	90.9
AIDE	86.0	93.8	65.8	95.2
SAFE	95.5	93.4	59.2	98.7
Effort	90.1	76.4	60.7	68.5

Method	GenImage			AIGCDB			Chameleon			CO-SPY		
	1	5	10	1	5	10	1	5	10	1	5	10
FSD	90.9	89.6	90.4	80.7	78.6	79.9	68.5	71.1	70.0	75.6	70.1	73.4
FTNet	78.7	82.8	83.8	72.9	78.2	80.5	65.2	73.7	67.2	41.8	60.0	68.4
Fleet	83.9	83.3	90.1	82.0	86.2	91.5	77.5	**88.2**	**90.9**	86.9	91.3	95.9
Ours	**92.3**	**95.2**	**93.7**	**92.9**	**94.9**	**93.7**	**84.0**	87.6	89.1	**96.7**	**98.4**	**97.3**

E.3 Complete Robustness Results

To assess the robustness of our method under common real-world degradations, we evaluate detection performance against a suite of image perturbations applied at varying intensities. Concretely, we consider two categories of perturbations: **Group A** covers compression and spatial degradations, including JPEG compression, Gaussian blur, Gaussian noise, and pixelation, while **Group B** covers color and photometric adjustments, such as contrast, saturation and brightness, with each of the

Table 10: Perturbation settings used in the robustness experiments. Seven perturbations are organized into two groups, each evaluated at five intensity levels with increasing strength.

Group	Perturbation	Parameter	L1	L2	L3	L4	L5
A	JPEG	quality	90	80	70	60	50
	Blur	Gaussian σ	0.5	1.0	1.5	2.0	3.0
	Noise	Gaussian σ	0.01	0.02	0.05	0.08	0.10
	Pixelation	block size	2	4	8	12	16
B	Contrast	factor	0.9	0.8	0.7	0.6	0.5
	Saturation	gain	0.8	0.6	0.4	0.2	0.0
	Brightness	factor	0.9	0.8	0.7	0.6	0.5

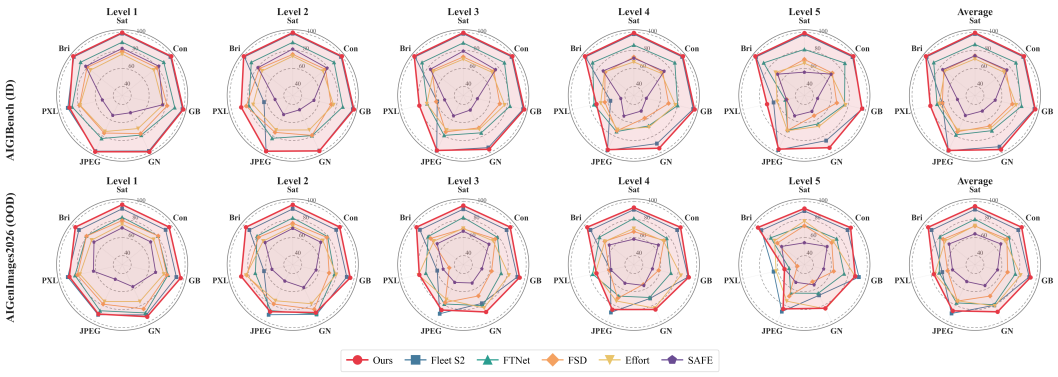

Figure 8: **Per-perturbation AP (%) under increasing perturbation intensity and their average.** Top: in-distribution results on AIGIBench; bottom: out-of-distribution results on AIGenImages2026. Our method consistently attains the largest radar area across all levels and both benchmarks.

seven perturbations applied at five increasing intensity levels. The specific operation parameters are detailed in Table 10, where perturbations are applied to the original-resolution RGB images prior to each method's own crop/resize; note that for JPEG a smaller quality value denotes a stronger perturbation.

Under the settings above, we evaluate our method against five competing detectors on both the in-distribution AIGIBench and the out-of-distribution AIGenImages2026 benchmarks, reporting per-perturbation AP across the five intensity levels in Figure 8. Across both benchmarks and all intensity levels, our method (red) consistently encloses the largest area on the radar plots, retaining the highest AP over nearly every perturbation type; while the AP of all detectors degrades as the intensity increases from L1 to L5, our method exhibits the slowest decline and remains dominant even at the strongest L5 setting. These results demonstrate that our approach is markedly more robust than existing detectors to a broad range of compression, spatial, and photometric degradations, and that this robustness generalizes well to the out-of-distribution setting. To provide an intuitive sense of the perturbation strengths, we further visualize representative examples of an image degraded by each perturbation in Figure 9.

E.4 Case Studies

We inspect our method's most confident errors on AIGIBench, shown in Figure 10. The errors fall into two modes: real images flagged as fake and synthetic images accepted as real. The former are dominated by heavily stylized or retouched real photographs whose over-regular textures and unnatural colors mimic generation cues; the latter are synthetic images that faithfully reproduce natural photographic statistics. We find both categories aggregate social-media content, where real photos undergo aggressive filtering and retouching while fakes are curated to look convincing. These



Figure 9: **Qualitative examples of a sample image under each perturbation at the L3 intensity levels, illustrating the visual severity of the applied degradations.**



Figure 10: **Representative failure cases on AIGIBench.** Top: real predicted as fake, typically stylized or post-processed photographs. Bottom: fake predicted as real, typically synthetic images mimicking natural photographic statistics.

borderline samples are therefore misclassified by most competing methods as well, indicating an intrinsic ambiguity of the data rather than a weakness specific to our approach.

To probe what drives our method making correct predictions, we visualize its attention maps on correctly classified samples in Figure 11. On generated images, RAIN concentrates on the synthesized foreground content, such as the dog's body and the building facades, where generation artifacts are most pronounced. On real images, its attention is diffuse and lacks such localized focus, and this pattern persists even on out-of-distribution cartoons that differ sharply in style from the training data. This contrast indicates that RAIN keys on authenticity evidence rather than on image content or style, consistent with the real-anchoring objective and supporting its generalization to unseen generators.

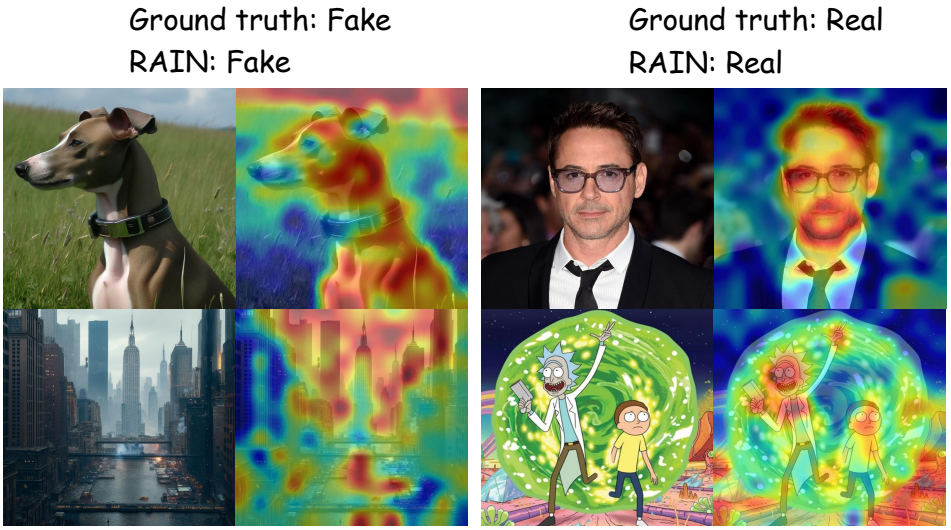

Figure 11: **Attention maps of RAIN on correctly classified samples.** RAIN attends to synthesized content regions in generated images while responding diffusely to real images, including out-of-distribution cartoons, consistent with a detector that keys on generation artifacts rather than content or style.

F Discussion

F.1 Relation to Real-Centric Methods in Traditional Deepfake Detection

Constraining the real distribution is an established idea in deepfake detection, motivated by the consensus that fake distributions form an open set while real distributions are relatively closed. Representative approaches include single-center loss (Li et al., 2021), which compresses real features toward one centroid; vector quantized modeling (Wang et al.), which discretizes the real manifold via a learnable codebook; and real distribution bias correction (Liu et al., 2026), which rectifies identified statistical deviations. All three approaches treat real images as geometric objects to be shaped, and all operate under single-source empirical risk minimization, where the loss cannot distinguish attributes co-varying with the label from genuine authenticity evidence. Consequently, the first reinforces whichever spurious cue the real subset happens to carry, the second struggles to cover the immense diversity of real-world content with a finite codebook, and the third requires every bias to be explicitly named beforehand.

Our method departs from this geometric paradigm by utilizing real images as an invariance reference rather than a shape to be constrained. Because real images belong to no specific generator, a detector capturing authenticity evidence must render stable judgments on the same real image regardless of the paired synthetic source. Multi-source environments render this consistency signal mathematically definable and differentiable, shifting the objective from shaping the geometry of real features to demanding cross-environment invariant predictions on them.

F.2 Alternative Attempts to Mitigate Shortcut Learning

Before arriving at RAIN, we explored four alternative strategies. Each failed for a structural reason, and together they delineate the design space that RAIN ultimately occupies.

Orthogonal decomposition via a biased teacher. We first trained a biased model to absorb shortcuts and then constrained the debiased model to be orthogonal to it. The decomposition is unidentifiable under single-source data because no supervisory signal separates shortcut features from authenticity evidence at the loss level. In practice, the biased model also captures genuine fake-side fingerprints, so orthogonalization pushes the debiased model away from valid forgery cues. Worse,

the debiased model can migrate toward a different shortcut that the biased model did not cover, merely swapping one spurious cue for another.

Adversarial removal of named shortcuts. We applied adversarial training to suppress specific attributes such as JPEG quality, content category, and visual style. This requires enumerating shortcuts in advance, which contradicts the open-set nature of the problem established in Section 3.2. Adversarial training against generator identity further weakens within-family transfer, since family-level fingerprints are legitimate evidence rather than shortcuts.

Cross-environment gradient consensus. We sought an optimization direction on which all environment gradients agree, hoping that consensus directions encode authenticity evidence. Because the fake side is an open set, generators with fundamentally different architectures produce genuinely different gradient signals, and no single direction is simultaneously optimal for all. Under the low-cost first-order approximation, the consensus direction degenerates toward the mean gradient, which is dominated by the easiest environment.

Parameter-level environment separation. We maintained an auxiliary environment discriminator throughout training to flag parameters carrying environment-specific information, then suppressed their learning rate to force the remaining parameters toward invariant features. By the time the discriminator reliably identifies shortcut parameters, the shortcut is already encoded in the weights. Erasing it would require a corrective gradient larger than the one that wrote it, making the procedure unstable.