The road reaches every place, the short cut only one: Self-Adversarial Shortcut Mitigation for AI-Generated Image Detection

Haifeng Zhang[1, 3], **Qinghui He**[1, 3], **Xiuli Bi**[2, 3], **Bo Liu**[2, 3,*], **Chi-Man Pun**[4], **Bin Xiao**[2, 3, 5,*]
[1]School of Computer Science and Technology, Chongqing University of Posts and Telecommunications
[2]School of Artificial Intelligence, Chongqing University of Posts and Telecommunications
[3]Chongqing Key Laboratory of Image Cognition, Chongqing, China
[4]Faculty of Information Science and Computing, University of Macau, Macao, China
[5]Jinan Inspur Data Technology Co., Ltd., Jinan, China
{D240201065, D250201011}@stu.cqupt.edu.cn, {bixl, boliu, xiaobin}@cqupt.edu.cn
cmpun@umac.mo

Abstract

With the rapid advancement of generative models, AI-generated images (AIGIs) have become increasingly realistic, posing new challenges for reliable detection. In this work, we revisit the root causes of poor generalization in AIGI detectors from the perspective of shortcut learning. We find that detectors often rely on spurious semantic cues or a single prominent artifact, causing the feature space to collapse toward low diversity and severely limiting generalization. To characterize this issue, we present the Generative Artifact Manifold Clustering (GAMC) hypothesis, an empirically supported characterization showing that although generative artifacts are multi-dimensional, different generators occupy distinct distributions along these dimensions, making detectors that rely on local artifact cues inherently fragile. To address this, we promote balanced discrimination across multiple artifact dimensions rather than reliance on a single dominant cue, and introduce the Semantic–Artifact Self-Adversarial Feature Learning (SA-AFL) framework. SA-AFL progressively disentangles semantics from artifacts and employs a self-adversarial mechanism to balance learning across artifact dimensions, enabling the model to retain richer and more shared generative features. As a result, SA-AFL reduces shortcut reliance and learns more transferable evidence for cross-generator detection. Comprehensive experiments on images from 23 generative models show that SA-AFL improves *ACC* by 6.84% and *AP* by 8.25% over state-of-the-art methods. Code is available at `https://github.com/oceanzhf/SA-AFL`.

1 Introduction

Generative AI models, particularly GANs [11, 18] and diffusion models [35, 14], have rapidly advanced, making high-quality image synthesis increasingly accessible [52]. While enabling applications in digital art and filmmaking, their misuse can spread misinformation and undermine trust in digital media [22, 23, 24, 25]. Therefore, effective AI-generated image (AIGI) detection has become an urgent challenge for both academia and industry [27, 26, 54].

Existing AIGI detectors often exploit salient artifacts in specific image dimensions, including frequency characteristics [9], noise patterns [29], color space [1], texture and gradient statistics [39], and local pixel relationships [40]. While effective when training and testing generators share similar

*Corresponding Authors.

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

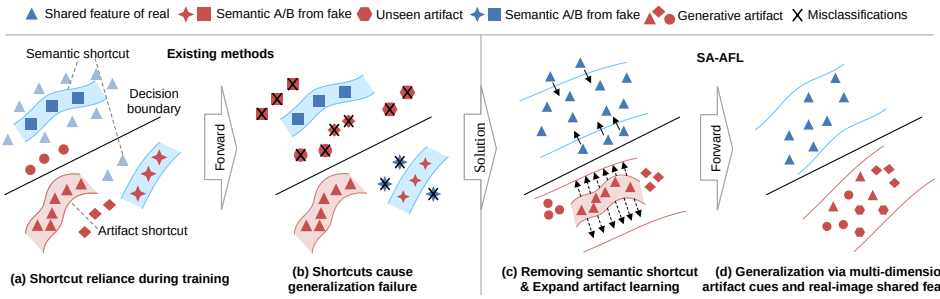


Figure 1: Illustration of shortcut reliance in AIGI detection. Our method not only eliminates semantic shortcuts but also prevents reliance on specific artifact shortcuts by jointly leveraging multi-dimensional generative artifacts.

distributions, such dimension-specific reliance limits generalization to unseen generators where these artifacts are weakened or shifted.

Recent works have found that richer and more diverse feature spaces are beneficial for improving detector generalization, and thus commonly employ large-scale pre-trained multimodal models to extract image features [33, 50, 52, 53]. Nevertheless, a growing body of research shows that such models often exhibit shortcut learning in downstream tasks, tending to rely on the most salient and easily separable spurious correlations in the data [10, 41]. For AIGI detection in complex real-world scenarios with continuously evolving generative models, shortcut learning induced by training data limited to a narrow range of scenes and generator types becomes particularly severe. This learning behavior progressively collapses the detector's discriminative space toward a single direction, causing generative artifacts in other dimensions to be overlooked.

We identify two major forms of shortcut learning in AIGI detection: shortcuts in generative artifacts and semantic content. With respect to generative artifact shortcuts, we propose the Generative Artifact Manifold Clustering (GAMC) hypothesis, which posits that, in deep representation spaces, generative artifacts exhibit flow-like aggregation along multiple dimensions and form distinct clusters. Different generative models occupy these clusters with markedly different distribution patterns, and detectors that overly rely on a single dimension suffer from severely limited cross-model generalization. For semantic shortcuts, multimodal pre-trained features encode broad semantics, style, and composition, which may spuriously align with real–fake labels under semantic distribution discrepancies. As shown in Fig. 1, such shortcuts can improve in-distribution accuracy but degrade cross-generator and cross-scene generalization, consistent with statistical learning theory that empirical risk minimization alone offers limited generalization guarantees [43].

These observations reveal a key challenge: semantic cues should be suppressed when they induce non-transferable label correlations, while generative artifacts should be preserved but balanced to avoid domination by a few source-specific dimensions. From this perspective, mitigating shortcut reliance requires shifting AIGI detection from single-cue dependence to multi-dimensional discriminative learning. To this end, we propose a Semantic–Artifact Self-Adversarial Feature Learning (SA-AFL) framework. SA-AFL decouples semantic content from generative artifacts, suppresses semantic shortcuts through adversarial disentanglement, and balances artifact learning through self-adversarial training. This encourages the detector to leverage complementary cues rather than irrelevant semantics or single dominant artifacts, improving generalization across scenes and generators.

In summary, our contributions are as follows:

- We identify shortcut learning in generated image detection and propose the Generative Artifact Manifold Clustering (GAMC) hypothesis, an empirically grounded characterization of the systematic biases of different generative models in representation space.
- We introduce a new perspective on generalizable AIGI detection, suggesting that generalization failures stem from over-reliance on locally salient discriminative dimensions rather than insufficient features. To the best of our knowledge, this is the first work to study cross-domain generalization in AIGI detection from a diversity learning perspective.

- Based on these insights, we propose a novel Semantic–Artifact Self-Adversarial Feature Learning (SA-AFL) framework, which progressively decouples semantic content and generative artifacts to mitigate shortcut reliance in AIGI detection.

2 Related Work

2.1 AI-Generated Image Detection

Early AIGI detectors mainly exploit spatial- and frequency-domain artifacts. CNNSpot [45] combines CNNs with data augmentation for GAN-image detection, while FreDect [9] and Liu et al. [29] focus on frequency artifacts and noise discrepancies. Other methods explore alternative domains, such as gradient artifacts in LGrad [39] and local pixel relationships in NPR [40]. Although effective on seen or similar generators, these methods generalize poorly under distribution shifts due to reliance on single or locally salient artifact cues.

Recent works improve generalization by leveraging large-scale pre-trained visual encoders, as in UnivFD [33], VIB-Net [52], and Effort [50]. However, they still struggle with shortcut reliance on spurious semantic correlations or prominent artifact cues.

2.2 Shortcut Learning and Its Mitigation

Shortcut learning has been widely observed in deep models, where models exploit predictive but non-causal cues instead of the full evidence [10]. Similar failures appear in large language models [41], Vision Transformers [31], and CLIP-based vision models [49, 44]. Existing mitigation methods usually target known nuisance factors, such as background regions, spurious saliency, or predefined semantic biases, and suppress or reweight these biased cues [53].

AIGI detection differs from these settings because generative artifacts are both essential evidence and potential shortcuts. Simply removing artifact-related cues may eliminate useful detection signals, while unconstrained learning can overfit to a few source-specific artifact dimensions. In the source-only setting, target generators and artifact-mode annotations are also unavailable, making predefined shortcut removal insufficient.

3 Preliminaries

3.1 Problem Background and Hypothesis

Current detection methods based on large-scale pre-trained multimodal models (e.g., CLIP) typically differentiate real and generated images by learning generic representations. We start from two observations. First, different generators vary in training data, architectures, objectives, and sampling procedures, and therefore tend to leave generator-dependent generative artifacts. Second, large-scale multimodal pre-trained representations provide generic visual features that encode semantics as well as visual cues such as style, texture, composition, and illumination.

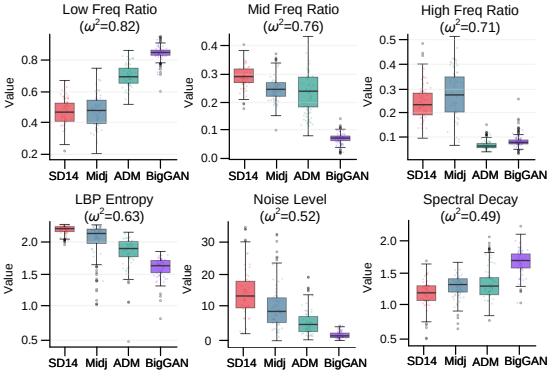


Figure 2: Artifact-feature distributions of four generative models on the six dimensions with the largest ANOVA effect sizes. Here, ω^2 denotes the bias-corrected one-way ANOVA effect size, where larger values indicate stronger generator-dependent distribution shifts.

Combining these observations, we propose the Generative Artifact Manifold Clustering (GAMC) hypothesis. Under semantic alignment, (i) images from different generators are distinguishable over a shared set of artifact-related dimensions. (ii) these discriminative artifact-related dimensions can be implicitly encoded by pre-trained multimodal representations, and (iii) consequently, even when semantic content is controlled, images from different generators remain separable in artifact-related subspaces. In this view, generative artifacts are not a single cue,

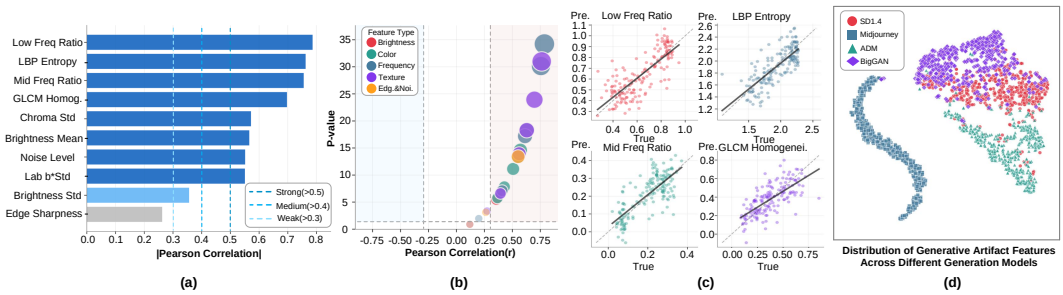


Figure 3: (a) Absolute Pearson correlations between CLIP-predicted and handcrafted artifact features. (b) Statistical significance of the correlations. (c) Predicted versus ground-truth handcrafted features from the linear probe. (d) t-SNE visualization of artifact representations, showing separated clusters for different generators.

but multi-dimensional traces that may aggregate into generator-specific manifold-like clusters in representation space. This leads to a shortcut risk: a source-only detector may rely on the locally most salient artifact dimensions of training generators, instead of learning transferable evidence across the artifact manifold. We stress that GAMC is an empirically supported characterization rather than a formal claim; Sec. 3.2 and Appendix A validate its three statements empirically.

3.2 Generative Artifact Distribution Analysis

We validated the three statements of GAMC using complementary analysis. To examine (i), we construct a semantically aligned set of 300 images generated by four representative models: Stable Diffusion v1.4, Midjourney, ADM, and BigGAN. We extract 25 handcrafted artifact statistics covering frequency, color, brightness/contrast, edge, texture, compression, and noise [9, 29, 1, 16]. We conduct one-way ANOVA with generator identity as the factor. The ANOVA p-value tests whether different generators have significantly different means, while ω^2 is a bias-corrected effect size that estimates the fraction of feature variance explained by generator identity. As shown in Fig. 2, 23 out of 25 dimensions are statistically significant at $p < 0.01$, with a mean ω^2 of 0.2699, especially in frequency-, brightness-, and texture-related statistics. These results support (i), showing that different generators are significantly distinguishable on a shared set of artifact-related dimensions. Full statistics are provided in Appendix A.

To examine (ii), we test whether these artifact-related dimensions are recoverable from CLIP representations. We freeze the CLIP image encoder and train a ridge-regression linear probe from CLIP features to handcrafted artifact features. Fig. 3(a)–(c) shows that the probe achieves a median absolute Pearson correlation of 0.551, and 20 out of 23 valid probed features are statistically significant at $p < 0.001$. This indicates that, beyond encoding semantic content, pre-trained multimodal representations also implicitly encode artifact-related dimensions.

To examine (iii), we train a real–fake classifier on CLIP features and visualize the learned artifact representations with t-SNE as a qualitative illustration. As shown in Fig. 3(d), samples from the four generators form separated clusters even under semantic alignment. This suggests that generator-dependent artifacts remain distinguishable in the feature space and tend to organize into generator-specific manifold-like clusters, consistent with GAMC. Together, these analyses support the shortcut risk: although generative artifacts are multi-dimensional, a source-only detector can overfit to a few locally salient artifact modes of the training generators, which may not transfer to unseen generators. This motivates balanced learning over multiple artifact dimensions rather than dependence on a single dominant artifact cue.

4 Methodology

Based on the analysis in Sec. 3, to achieve this, as illustrated in Fig. 4, we propose a Semantic–Artifact Self-Adversarial Feature Learning (SA-AFL) framework that progressively disentangles semantic content and generative artifacts through a two-stage process.

Formally, let the feature set extracted by a pre-trained visual encoder be $\{(x_i, y_i)\}_{i=1}^{N}$, where $x_i \in \mathbb{R}^d$ represents the visual feature of the i-th sample and $y_i \in \{0, 1\}$ is its ground-truth label. SA-AFL first

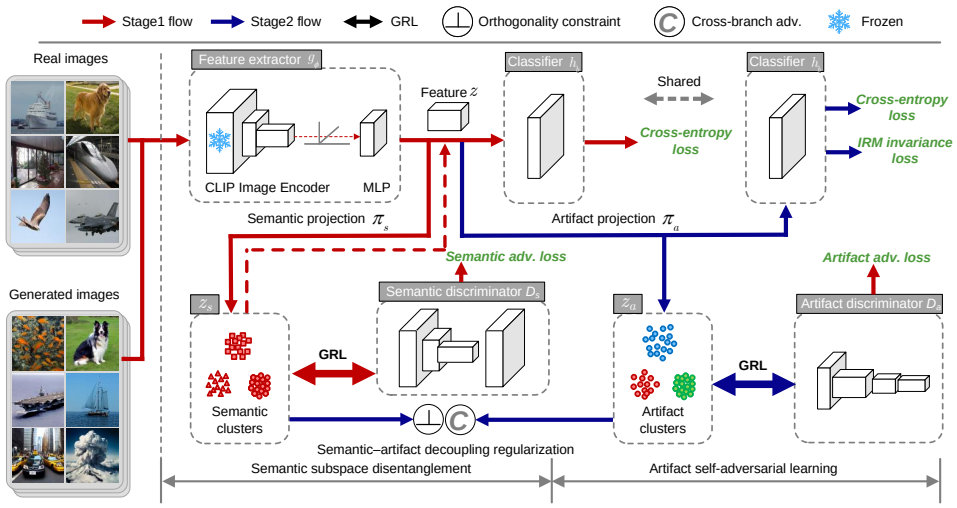


Figure 4: Overview of SA-AFL. Stage I suppresses semantic shortcuts through semantic adversarial disentanglement. Stage II performs artifact self-adversarial learning to reduce reliance on dominant artifact dimensions and promote multi-dimensional artifact discrimination.

extracts intermediate features with a shared backbone g_ϕ:

$$z_i = g_\phi(x_i) \in \mathbb{R}^D, \tag{1}$$

a binary classifier h_b is then applied to produce the real-fake prediction:

$$\hat{y}_i = h_b\left(z_i\right), \tag{2}$$

which is optimized using the standard cross-entropy loss L_{clf}. To disentangle semantics and artifacts, we project z_i into two subspaces:

$$z_i^s = \pi_s\left(z_i\right), z_i^a = \pi_a\left(z_i\right), \tag{3}$$

and introduce two branches equipped with Gradient Reversal Layers (GRL).

Semantic branch: A semantic discriminator D_s takes $\mathrm{GRL}_{\lambda_s}\left(z_i^s\right)$ and outputs a semantic-cluster prediction $\hat{s}_i$, thereby suppressing semantic shortcuts.

Artifact branch: An artifact discriminator D_a takes $\mathrm{GRL}_{\lambda_a}\left(z_i^a\right)$ and outputs an artifact-cluster prediction $\hat{a}_i$, preventing the artifact subspace from being dominated by locally salient artifact dimensions.

We further introduce a cross-branch predictor $h_{a\to s}$ and an orthogonality constraint between the two subspaces to enhance disentanglement. Training proceeds in two stages: Stage I performs semantic disentanglement. Stage II applies self-adversarial learning in the artifact space to achieve comprehensive multi-dimensional artifact discrimination.

4.1 Semantic Subspace Disentanglement

The goal of the first stage is to suppress semantic-content shortcuts, reducing the model's reliance on content-related discriminative cues.

Semantic Clustering. We first merge the features of real and generated images $\{x_i\}_{i=1}^N$ and apply K-means clustering in the feature space to obtain pseudo-semantic labels, where the cluster number K_s is adaptively selected on the training set:

$$\{s_i\}_{i=1}^N = \text{K-means}\big(\{x_i\}_{i=1}^N, K_s\big), \quad s_i \in \{1, \ldots, K_s\}, \tag{4}$$

where s_i denotes the pseudo-semantic label of sample i.

Semantic Adversarial Learning. To remove semantic-discriminative information, we adopt adversarial training via a Gradient Reversal Layer (GRL). The GRL operator R_λ is defined as

$$R_\lambda(u) = u, \qquad \frac{\partial R_\lambda(u)}{\partial u} = -\lambda I, \tag{5}$$

which preserves the forward activations but reverses the gradients passed to the encoder. A semantic discriminator D_s takes $R_{\lambda_s}(z_i^s)$, where z_i^s is defined in Eq. (3), as input and uses the cluster labels $\{s_i\}$ as supervision, and is trained with the cross-entropy loss:

$$\mathcal{L}_{s\text{-CE}} = -\frac{1}{N}\sum_{i=1}^{N}\log p(s_i \mid R_{\lambda_s}(z_i^s)). \tag{6}$$

Through the GRL, gradients are reversed when back-propagated to the encoder, encouraging pseudo-semantic-invariant representations.

After Stage I training, semantic-correlated feature components are effectively suppressed, shifting the detector's decision basis toward cleaner generative artifact cues.

4.2 Artifact Self-Adversarial Learning

The second stage further suppresses generative artifact shortcuts in the de-semanticized feature space, with the core objective of preventing the model from relying on a single artifact dimension.

Conditional Residualization and Artifact Clustering. After the joint real–fake classification and semantic adversarial learning in Stage I, the resulting representations are mainly driven by real–fake discriminative cues. To further reduce the interference of residual semantic information in multi-dimensional artifact clustering, we first perform cluster-wise mean removal within each semantic cluster to obtain residual features:

$$\tilde{z}_i = z_i - \frac{1}{|S_{s_i}|}\sum_{j\in S_{s_i}} z_j, \tag{7}$$

where $S_k = \{j \in \{1,\dots,N\} \mid s_j = k\}$ denotes the index set of samples in semantic cluster k, and S_{s_i} is the cluster containing sample i.

We then apply K-means clustering to the residual features $\tilde{z}_i$ over the set of generated images, yielding artifact clusters, where the cluster number K_a is adaptively selected on the training set:

$$\{a_i\}_{i\in\mathcal{F}} = \text{K-means}\left(\{\tilde{z}_i\}_{i\in\mathcal{F}}, K_a\right), \quad a_i \in \{1,\dots,K_a\}, \tag{8}$$

where a_i denotes the artifact cluster assignment. These clusters capture diverse dimensions of generative artifacts and provide structured supervision for subsequent artifact adversarial learning. To verify that these clusters capture generative artifacts, we conduct controlled artifact/semantic interventions, with results provided in Appendix B.

Multi-dimensional Artifact Adversarial. In the de-semanticized artifact feature space, we further suppress the model's reliance on any single artifact dimension. Specifically, a K_a-way artifact discriminator D_a takes artifact subspace features z_i^a as input and predicts artifact cluster labels, while adversarial training is implemented via a Gradient Reversal Layer (GRL):

$$\mathcal{L}_a = -\frac{1}{|\mathcal{F}|}\sum_{i\in\mathcal{F}}\log p\left(a_i \mid R_{\lambda_a}(z_i^a)\right). \tag{9}$$

This adversarial mechanism has a self-balancing effect: when the encoder overemphasizes a particular artifact dimension, the discriminator rapidly exploits it, and the reversed gradients subsequently suppress the encoder's dependence on that dimension. Note that the objective removes *generator-specific* reliance rather than artifact information per se: it suppresses excess predictability of any single artifact mode while preserving the shared artifact cues that transfer across generators.

Semantic–Artifact Decoupling Regularization. To ensure that semantic and artifact subspaces encode mutually independent information, we introduce multi-level decoupling regularization that constrains both representation learning and decision behavior.

We first enforce orthogonality between the semantic subspace z^s and the artifact subspace z^a:

$$L_{orth} = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{(z_i^s)^\top z_i^a}{\|z_i^s\|\,\|z_i^a\|}\right)^2, \tag{10}$$

encouraging the two subspaces to capture independent information.

Table 1: Cross-generator generalization performance (*ACC%*). All methods are trained on the GenImage training set (SDV1.4) and then evaluated on both the GenImage and ForenSynths test sets (Protocol 1). **Bold** indicates the best result in each column. Underline indicates the second best.

Method	Generative Adversarial Networks						Diffusion Models							*ACC*
	ProGAN	StyleGAN	CycleGAN	StarGAN	BigGAN	GauGAN	SDV1.4	VQDM	SDV1.5	GLIDE	ADM	Midjourney	Wukong	**Avg.**
Fusing[17] (ICIP2022)	51.20	50.20	52.40	58.20	53.50	49.32	99.90	64.20	99.91	57.50	51.30	52.30	99.90	64.60
Lgrad[39] (CVPR2023)	61.61	61.43	60.74	50.17	48.82	49.70	99.12	50.93	99.05	64.24	53.00	76.34	97.53	67.13
DIRE[46] (ICCV2023)	52.93	52.49	52.12	54.90	52.75	50.73	98.08	**98.75**	98.26	**94.85**	**96.60**	**95.87**	98.42	76.67
UnivFD[33] (CVPR2023)	61.75	66.50	65.55	67.20	65.65	60.20	86.60	60.70	87.80	78.20	58.85	75.90	78.20	70.24
NPR[40] (CVPR2024)	60.30	58.00	67.20	73.20	59.20	52.00	**100.00**	68.30	99.90	89.70	73.00	82.30	**100.00**	75.62
DRCT[2] (ICML2024)	76.83	71.58	94.51	63.13	82.68	78.39	95.01	90.03	94.41	89.18	79.42	88.50	94.68	84.49
CLIPping[20]	88.78	80.69	88.48	90.82	89.57	85.94	96.07	79.35	95.48	85.00	70.14	77.66	88.86	85.91
CO-SPY[5] (CVPR2025)	77.84	68.88	69.65	67.48	76.75	77.57	88.88	78.53	94.25	80.78	68.67	78.59	88.93	78.21
Effort[50] (ICML2025)	94.00	75.00	88.50	**95.50**	**93.50**	80.00	**100.00**	89.50	**100.00**	85.00	72.00	78.00	99.50	88.50
VIB-Net[52] (CVPR2025)	89.70	74.10	88.60	80.70	91.20	87.15	99.55	89.35	99.20	74.25	73.85	88.05	98.25	87.23
Ours	**96.25**	**89.29**	**97.84**	93.12	89.58	**90.03**	98.57	95.51	98.52	90.93	86.47	78.63	98.52	**92.56**

To further prevent semantic leakage into artifact representations, we introduce a cross-branch adversarial loss by predicting semantic cluster labels from the artifact subspace via GRL:

$$\mathcal{L}_{cross} = -\frac{1}{N}\sum_{i=1}^{N} \log p\left(s_i \mid R_{\lambda_{cs}}\left(z_i^a\right)\right). \tag{11}$$

Finally, inspired by Invariant Risk Minimization (IRM), we enforce consistent real–fake predictions across different semantic clusters:

$$L_{inv} = \mathrm{Var}\left(\left\{\frac{1}{|S_k \cap \mathcal{F}|}\sum_{i\in S_k\cap\mathcal{F}} [h_b(z_i)]_1\right\}_{k=1}^{K_s}\right). \tag{12}$$

This constraint acts directly on the decision layer, preventing systematic prediction bias induced by semantic variations.

5 Experiments

5.1 Experimental Setup

Datasets. To comprehensively evaluate our method, we adopt three protocols. **Protocol 1:** Train on the SDV1.4 subset of GenImage [57] and test on other GenImage diffusion models and the GAN-based ForenSynths dataset [45]. **Protocol 2:** Following prior work [40, 15], train on four ProGAN subsets (car, cat, chair, horse) of ForenSynths and evaluate on both GenImage and ForenSynths. Together, Protocols 1 and 2 form a cross-model evaluation setting covering both diffusion and GAN families. **Protocol 3:** Train on the Holmes [55] training split and evaluate on newly released diffusion models (FLUX [21], PixArt-XL [3], SD3.5 [8]) and autoregressive models (LlamaGen [38], Infinity [13], Janus-Pro [4], VAR [42], Janus [47], Show-o [48]).

Implementation Details. We implement our method in PyTorch 2.7 and train on an NVIDIA RTX A6000 with CLIP ViT-L/14. For clustering, we use Davies–Bouldin-index-guided K-means to adaptively select K_s and K_a on the training set, with $\mathcal{K}_s = \{20, \ldots, 200\}$ and $\mathcal{K}_a = \{2, \ldots, 20\}$ as the candidate ranges. Training uses AdamW with weight decay 1×10^{-2}, learning rate 2×10^{-4}, cosine annealing, and a batch size of 256. The adversarial weight λ uses a warm-up value of 0.01 for 3 epochs, followed by linear growth. The cross-branch and IRM coefficients are set to $\lambda_{cs} = 0.5$ and $\lambda_{inv} = 0.1$, respectively. All comparison methods are evaluated under the same three protocols.

5.2 Performance comparisons

Diffusion-trained Performance Evaluation. Following Protocol 1, all methods are trained on the SDV1.4 subset of GenImage and evaluated on unseen diffusion and GAN generators. As shown in Tabs. 1 and 2, SA-AFL achieves the best average performance, with 92.56% *ACC* and 97.73% *AP*. Compared with UnivFD, which directly classifies CLIP features, SA-AFL improves *ACC* and *AP* by 22.32 and 19.60 percentage points, respectively, showing the benefit of semantic–artifact decoupling and balanced artifact learning. SA-AFL also outperforms recent CLIP-based detectors Effort and VIB-Net by 4.06 and 5.33 *ACC* points, respectively, indicating that mitigating both semantic and dominant-artifact shortcuts improves cross-generator generalization.

Table 2: Performance metrics (*AP*) of various methods trained on images generated by SDV1.4. Generative models other than SDv1.4 are treated as out-of-domain settings for evaluating generalization, with GAN-based models being a particularly challenging case.

Method	Generative Adversarial Networks						Diffusion Models							*AP*
	ProGAN	StyleGAN	CycleGAN	StarGAN	BigGAN	GauGAN	SDV1.4	VQDM	SDV1.5	GLIDE	ADM	Midjourney	Wukong	**Avg.**
Fusing[17] (ICIP2022)	67.63	67.90	87.79	91.20	69.37	43.08	99.80	84.60	97.98	94.20	69.30	81.20	99.90	81.07
Lgrad[39] (CVPR2023)	83.59	82.74	90.24	**99.19**	47.51	49.25	99.94	56.34	99.92	58.52	58.52	91.06	99.72	78.20
DIRE[46] (ICCV2023)	49.68	50.40	49.89	48.46	51.52	50.55	99.83	**99.99**	99.85	**99.13**	**99.39**	99.25	99.88	76.75
UnivFD[33] (CVPR2023)	58.37	59.20	67.22	59.93	80.21	70.90	96.20	78.40	96.26	93.73	72.25	92.08	90.98	78.13
NPR[40] (CVPR2024)	83.30	82.70	94.90	97.30	72.00	66.00	**100.00**	86.30	99.97	95.80	94.70	95.50	**100.00**	89.88
DRCT[2] (ICML2024)	91.03	79.44	98.68	96.29	93.51	86.62	99.65	96.54	99.49	94.64	88.96	97.03	99.37	93.94
CLIPping[20]	88.54	77.23	88.44	89.82	85.33	81.56	93.97	77.17	93.10	87.44	68.00	77.34	86.52	84.19
CO-SPY[5] (CVPR2025)	82.26	69.95	68.28	71.40	78.41	78.74	95.34	88.65	98.63	91.63	82.42	97.81	99.93	84.88
Effort[50] (ICML2025)	98.58	95.61	92.60	98.43	**97.56**	89.93	**100.00**	99.71	**100.00**	97.38	91.00	98.68	**100.00**	96.88
VIB-Net[52] (CVPR2025)	96.59	84.31	98.44	97.60	97.17	**96.94**	**100.00**	97.00	99.97	97.13	95.49	97.81	99.93	96.80
Ours	**99.39**	**97.81**	**99.27**	98.94	95.35	87.43	99.92	98.73	99.75	98.31	96.24	**99.69**	99.66	**97.73**

Table 3: Cross-generator generalization performance (*ACC%*). Evaluation on the Protocol 2. For a fair comparison, the classification threshold for all models is set to 0.5.

Method	Generative Adversarial Networks						Diffusion Models							*ACC*
	ProGAN	StyleGAN	CycleGAN	StarGAN	BigGAN	GauGAN	SDV1.4	VQDM	SDV1.5	GLIDE	ADM	Midjourney	Wukong	**Avg.**
UnivFD[33] (CVPR2023)	**100.00**	79.00	93.50	87.00	97.50	98.50	55.50	**91.50**	54.00	66.50	74.00	48.00	56.50	77.04
NPR[40] (CVPR2024)	99.60	90.80	81.20	**99.00**	73.00	62.00	79.00	61.00	**86.00**	82.00	71.00	**82.00**	81.00	80.58
CLIPping[20]	99.77	89.50	87.26	98.09	94.27	95.63	71.29	83.26	72.48	84.29	75.45	57.27	70.01	82.96
Effort[50] (ICML2025)	**100.00**	87.50	92.50	96.50	**98.50**	**100.00**	72.00	86.50	74.00	83.50	65.50	54.00	76.00	83.58
VIB-Net[52] (CVPR2025)	99.50	83.00	93.50	93.00	92.50	**100.00**	70.00	86.00	70.00	62.00	69.00	60.50	74.00	81.00
Ours	99.81	**94.19**	**96.33**	96.17	95.50	94.06	**79.65**	90.43	79.10	**88.67**	**81.78**	62.32	**83.88**	**87.84**

Table 4: Performance metrics (*AP*) of various methods trained on images generated by ProGAN.

Method	Generative Adversarial Networks						Diffusion Models							*AP*
	ProGAN	StyleGAN	CycleGAN	StarGAN	BigGAN	GauGAN	SDV1.4	VQDM	SDV1.5	GLIDE	ADM	Midjourney	Wukong	**Avg.**
UnivFD[33] (CVPR2023)	**100.00**	96.20	**99.99**	99.87	99.53	**100.00**	64.32	**98.10**	57.17	81.97	84.97	42.48	69.82	84.19
NPR[40] (CVPR2024)	**100.00**	**99.30**	98.20	**100.00**	76.10	70.10	**97.50**	66.10	85.20	62.70	77.10	**86.30**	90.50	85.32
CLIPping[20]	99.76	89.27	96.68	97.16	90.48	91.96	77.01	87.44	76.47	82.30	78.73	55.86	81.02	84.93
Effort[50] (ICML2025)	**100.00**	98.22	95.46	99.96	**99.99**	**100.00**	91.01	94.94	**92.25**	94.69	75.42	65.48	90.44	92.14
VIB-Net[52] (CVPR2025)	99.81	89.59	99.32	96.45	98.06	**100.00**	76.98	91.87	77.92	72.98	78.25	66.91	83.34	87.04
Ours	99.99	98.63	99.20	99.71	99.43	99.62	91.87	97.36	91.80	**96.32**	**93.23**	77.20	**94.45**	**95.29**

Table 5: Evaluation on the Protocol 3. The test set includes the latest autoregressive and diffusion models.

Method	Janus		Janus-Pro-1B		Janus-Pro-7B		Show-o		LlamaGen		Infinity		VAR		PixArt-XL		SD35-L		FLUX		Mean	
	ACC	*AP*	*ACC*	*AP*	*ACC*	*AP*	*ACC*	*AP*	*ACC*	*AP*	*ACC*	AP	*ACC*	*AP*	*ACC*	*AP*	*ACC*	*AP*	*ACC*	*AP*	*ACC*	*AP*
CNNspot	70.00	86.00	70.90	85.80	85.00	93.60	72.20	86.00	61.90	71.40	86.80	94.60	59.90	75.00	78.20	90.10	63.80	81.10	81.10	92.00	72.90	85.60
FreDect	66.68	63.90	78.37	75.57	78.53	76.15	71.50	72.32	76.69	84.25	79.83	84.65	63.79	65.62	72.49	74.60	68.07	69.25	75.76	90.46	73.17	75.68
Fusing	92.58	96.22	95.56	97.52	95.95	97.77	96.11	97.97	96.08	97.97	86.33	87.99	86.13	87.97	95.86	97.85	90.80	92.80	**95.72**	97.84	93.11	95.19
UnivFD	87.60	97.80	96.90	99.50	96.40	99.50	85.90	97.40	93.10	98.60	79.20	96.20	64.30	85.90	75.70	94.40	87.80	97.80	69.60	91.40	83.60	95.90
CLIPping	84.62	84.29	97.03	96.66	95.94	95.68	92.21	91.89	96.30	96.02	90.47	90.21	67.59	67.31	88.85	88.53	96.57	96.36	84.22	84.06	89.38	89.10
NPR	51.20	55.90	69.50	75.10	73.90	77.90	93.70	99.60	93.50	99.40	93.80	99.90	85.90	91.20	93.40	99.10	91.60	97.70	93.60	**99.50**	84.00	89.50
LaRE	70.80	99.30	74.70	97.50	95.60	99.70	80.00	99.00	91.60	99.60	77.90	99.60	**98.80**	**100.00**	82.20	99.70	94.10	99.50	84.30	99.00	85.00	99.30
VIB-Net	92.50	99.61	97.00	99.80	97.50	99.92	94.50	99.73	**99.50**	**99.99**	97.50	99.90	71.50	93.21	91.00	98.75	98.00	99.87	88.50	98.79	92.75	98.96
Effort	96.00	99.50	97.50	99.76	96.50	99.73	93.50	99.42	98.50	99.95	97.50	**99.99**	80.50	96.33	95.00	99.74	98.00	**99.97**	88.00	98.42	94.10	99.28
Ours	**98.51**	**99.73**	**99.26**	**99.91**	**98.89**	**99.94**	**99.07**	**99.93**	98.80	99.89	**99.42**	99.97	94.50	98.60	**98.35**	**99.82**	**99.18**	99.94	94.85	99.03	**98.08**	**99.67**

GAN-trained Performance Evaluation. Under Protocol 2, all methods are trained on ProGAN images and evaluated on unseen GAN and diffusion generators. As shown in Tabs. 3 and 4, most detectors perform well within the GAN family but degrade on diffusion models. SA-AFL achieves 87.84% *ACC* and 95.29% *AP*, improving over VIB-Net by 6.84 *ACC* points and 8.25 *AP* points. These results show that mitigating both semantic interference and dominant-artifact shortcuts improves generalization from GAN-trained detectors to broader generative models.

Performance Evaluation on the Latest Unseen Generative Models. Under Protocol 3, we evaluate detectors on recently released diffusion and autoregressive generators unseen during training. As shown in Tab. 5, SA-AFL achieves the strongest mean performance, with 98.08% *ACC* and 99.67% *AP*, improving over UnivFD, Effort, and VIB-Net by 14.48, 3.98, and 5.13 ACC points, respectively. It also shows clear gains over Effort on challenging generators such as FLUX and VAR, with *ACC* improvements of 6.85 and 14.00 points. Overall, results across all three protocols confirm the effectiveness of mitigating semantic and artifact shortcuts for cross-generator generalization.

5.3 Ablation Study

To assess the contribution of each component, we perform ablation studies under Protocol 1 (Tab. 6). The baseline achieves 84.55% *ACC*. Adding semantic subspace decoupling (stage 1) improves *ACC*

Table 6: Ablation study on the effectiveness of each component.

CLIP:ViT	Sem. Decoup.	Art. Adv.	$\mathcal{L}_{cs}$	$\mathcal{L}_{inv}$	$\mathcal{L}_{orth}$	Avg.
✓	×	×	×	×	×	84.55
✓	✓	×	×	×	×	88.28
✓	✓	✓	×	×	×	90.81
✓	✓	✓	×	✓	✓	91.25
✓	✓	✓	✓	×	✓	91.54
✓	✓	✓	✓	✓	×	91.83
✓	✓	✓	✓	✓	✓	**92.56**

to 88.28% (+3.73%), confirming its effectiveness in reducing semantic shortcuts. Incorporating artifact-subspace self-adversarial learning further boosts *ACC* to 90.81%, indicating reduced reliance on dominant artifact cues.

Introducing multi-dimensional regularization (cross-branch adversarial, orthogonality, and semantic invariance) improves *ACC* to 91.25–91.83% by mitigating semantic interference. Combining all components yields the best performance at 92.56%. Overall, semantic decoupling and artifact self-adversarial learning contribute the most, while multi-dimensional regularization provides complementary gains.

Empirical Analysis of Shortcut Mitigation. Beyond the component ablation, we directly diagnose whether SA-AFL suppresses shortcut information in the learned representation. Linear probes show that SA-AFL reduces semantic-cluster predictability from 0.909 to 0.625 and generator-ID predictability from 0.735 to 0.584, while artifact clusters become more compact and balanced after self-adversarial learning. A controlled semantic-conflict test further reduces the shortcut error from 30.4% to 12.3%. These results indicate that the performance gain comes from reduced semantic and generator-specific shortcut reliance. Full diagnostics are provided in Appendix C.2.

Clustering choices and K sensitivity. Since SA-AFL uses semantic and artifact clusters as pseudo-labels, we examine whether its effectiveness depends on a fragile clustering configuration. Across wide ranges of K_s and K_a, data-driven automatic K selection, and K-free clustering methods, SA-AFL maintains strong performance without requiring precise manual tuning. Replacing K-means with Spectral Clustering, GMM, or UMAP+K-means also preserves strong results. In particular, UMAP+K-means further improves the overall ACC, suggesting that the gain comes from the self-adversarial shortcut-mitigation mechanism rather than a specific clustering algorithm. Full results are provided in Appendix C.4 and Appendix C.5.

Backbone transferability and efficiency. SA-AFL also shows practical transferability and modest overhead. It consistently improves over the corresponding baseline across CLIP ViT-B/16, MetaCLIP, DINOv2, and CLIP ViT-L/14, with gains of 3.28–8.01 *ACC* points. Since clustering and adversarial heads are training-time components, the additional cost is limited: the two clustering steps take 17.41s and 3.12s, and inference remains comparable to other CLIP-based detectors at 12.3 ms/image. Details are given in Appendix C.6 and Appendix C.7.

5.4 Robustness Evaluation

In real-world scenarios, images often undergo post-processing operations, making robustness essential alongside cross-generator generalization. Under Protocol 1, we first evaluate JPEG compression and Gaussian blur in Fig. 5. We further test resizing, additive Gaussian noise, color perturbations, and multi-step post-processing pipelines in Appendix C.8. SA-AFL consistently outperforms competing methods across these perturbations. In particular, under brightness, contrast, and hue shifts, it exceeds the strongest baseline by 12.22–12.72% *ACC*. These results show that the robustness gain is not limited to JPEG/blur, but transfers to broader image-transmission distortions.

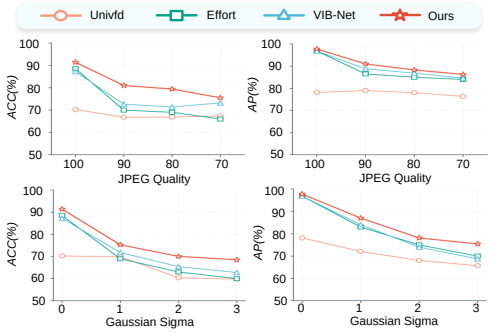


Figure 5: Robustness under Protocol 1 perturbations: JPEG compression (top) and Gaussian blur (bottom).

6 Conclusion

In this paper, we revisit the generalization challenge of AI-generated image detection from the perspective of shortcut learning. Our analysis shows that detectors built on large-scale pre-trained encoders often rely on spurious semantic cues or overly salient artifact dimensions, making them difficult to generalize to unseen generators and diverse scenes. To address this, we propose the SA-AFL framework, which progressively decouples semantic content from generative artifacts and employs self-adversarial learning to achieve balanced discrimination across multiple artifact dimensions, suppressing semantic shortcuts while avoiding over-reliance on any single artifact cue. We further uncover a critical insight—a hidden pitfall in AIGI detection: although generative artifacts are inherently multi-dimensional, different generative models distribute along these dimensions in distinct ways. Thus, detectors that rely on locally salient cues fail to generalize when those cues become less prominent. Extensive experiments demonstrate that SA-AFL consistently outperforms existing methods, validating the effectiveness of mitigating shortcut reliance for improved generalization in AIGI detection.

Acknowledgments and Disclosure of Funding

We thank the anonymous reviewers and the area chair for their constructive comments, which helped us improve this paper. Funding in direct support of this work: the National Natural Science Foundation of China under Grant 62376064, Grant 62536002, Grant 62561160098, and Grant 62576060; the National Natural Science Foundation of China Joint Key Project under Grant U24B20182 and Grant U24B20173; the Natural Science Foundation of Chongqing under Grant CSTB2025YITP-QCRCX0074; and the Chongqing Graduate Research and Innovation Project under Grant CYB25261. Competing interests: the authors declare no financial relationships outside the affiliations listed above that could be perceived as influencing this work.

References

[1] Keshigeyan Chandrasegaran, Ngoc-Trung Tran, Alexander Binder, and Ngai-Man Cheung. Discovering transferable forensic features for cnn-generated images detection. In *European Conference on Computer Vision*, pages 671–689. Springer, 2022.

[2] Baoying Chen, Jishen Zeng, Jianquan Yang, and Rui Yang. Drct: Diffusion reconstruction contrastive training towards universal detection of diffusion generated images. In *Forty-first International Conference on Machine Learning*, 2024.

[3] Junsong Chen, Chongjian Ge, Enze Xie, Yue Wu, Lewei Yao, Xiaozhe Ren, Zhongdao Wang, Ping Luo, Huchuan Lu, and Zhenguo Li. Pixart-σ: Weak-to-strong training of diffusion transformer for 4k text-to-image generation. In *European Conference on Computer Vision*, pages 74–91. Springer, 2024.

[4] Xiaokang Chen, Zhiyu Wu, Xingchao Liu, Zizheng Pan, Wen Liu, Zhenda Xie, Xingkai Yu, and Chong Ruan. Janus-pro: Unified multimodal understanding and generation with data and model scaling. *arXiv preprint arXiv:2501.17811*, 2025.

[5] 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*, pages 13455–13465, 2025.

[6] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8789–8797, 2018.

[7] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.

[8] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning*, 2024.

[9] 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*, pages 3247–3258. PMLR, 2020.

[10] Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673, 2020.

[11] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014. URL https://proceedings.neurips.cc/paper_files/paper/2014/file/5ca3e9b122f61f8f06494c97b1afccf3-Paper.pdf.

[12] Shuyang Gu, Dong Chen, Jianmin Bao, Fang Wen, Bo Zhang, Dongdong Chen, Lu Yuan, and Baining Guo. Vector quantized diffusion model for text-to-image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10696–10706, 2022.

[13] Jian Han, Jinlai Liu, Yi Jiang, Bin Yan, Yuqi Zhang, Zehuan Yuan, Bingyue Peng, and Xiaobing Liu. Infinity: Scaling bitwise autoregressive modeling for high-resolution image synthesis. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 15733–15744, 2025.

[14] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[15] Yonghyun Jeong, Doyeon Kim, Youngmin Ro, and Jongwon Choi. Frepgan: robust deepfake detection using frequency-level perturbations. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pages 1060–1068, 2022.

[16] Zexi Jia, Chuanwei Huang, Yeshuang Zhu, Hongyan Fei, Xiaoyue Duan, Zhiqiang Yuan, Ying Deng, Jiapei Zhang, Jinchao Zhang, and Jie Zhou. Secret lies in color: Enhancing ai-generated images detection with color distribution analysis. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 13445–13454, 2025.

[17] Yan Ju, Shan Jia, Lipeng Ke, Hongfei Xue, Koki Nagano, and Siwei Lyu. Fusing global and local features for generalized ai-synthesized image detection. In *2022 IEEE International Conference on Image Processing (ICIP)*, pages 3465–3469. IEEE, 2022.

[18] Tero Karras. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017.

[19] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410, 2019.

[20] Sohail Ahmed Khan and Duc-Tien Dang-Nguyen. Clipping the deception: Adapting vision-language models for universal deepfake detection. In *Proceedings of the 2024 International Conference on Multimedia Retrieval*, pages 1006–1015, 2024.

[21] Black Forest Labs. Flux. https://github.com/black-forest-labs/flux, 2024.

[22] Shuaibo Li, Shibiao Xu, Wei Ma, and Qiu Zong. Image manipulation localization using attentional cross-domain cnn features. *IEEE Transactions on Neural Networks and Learning Systems*, 34(9):5614–5628, 2021.

[23] Shuaibo Li, Wei Ma, Jianwei Guo, Shibiao Xu, Benchong Li, and Xiaopeng Zhang. Unionformer: Unified-learning transformer with multi-view representation for image manipulation detection and localization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12523–12533, 2024.

[24] Shuaibo Li, Zhaohu Xing, Hongqiu Wang, Pengfei Hao, Xingyu Li, Zekai Liu, and Lei Zhu. Toward medical deepfake detection: A comprehensive dataset and novel method. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 626–637. Springer, 2025.

[25] Shuaibo Li, Pengfei Hao, Hongtao Wu, Jianfeng Dong, Ping Li, Xiaohong Liu, and Lei Zhu. Hermes: An evidence-driven agentic framework for trustworthy and explainable ai-generated video detection. In *Forty-third International Conference on Machine Learning*, 2026.

[26] Shuaibo Li, Yijun Yang, Zhaohu Xing, Hongqiu Wang, Pengfei Hao, Xingyu Li, Zekai Liu, Qing Zhang, and Lei Zhu. Synerdetect: Hierarchical synergistic learning for generalizable ai-generated image detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 6405–6414, 2026. doi: 10.1609/aaai.v40i8.37568.

[27] Shuaibo Li, Laixin Zhang, Wei Ma, Jianwei Guo, Shibiao Xu, Zhijie Qiu, and Hongbin Zha. Realnet: Efficient and unsupervised detection of ai-generated images via real-only representation learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 38889–38898, 2026.

[28] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014.

[29] Bo Liu, Fan Yang, Xiuli Bi, Bin Xiao, Weisheng Li, and Xinbo Gao. Detecting generated images by real images. In *European Conference on Computer Vision*, pages 95–110. Springer, 2022.

[30] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of the IEEE international conference on computer vision*, pages 3730–3738, 2015.

[31] Chong Ma, Lin Zhao, Yuzhong Chen, Lei Guo, Tuo Zhang, Xintao Hu, Dinggang Shen, Xi Jiang, and Tianming Liu. Rectify vit shortcut learning by visual saliency. *IEEE Transactions on Neural Networks and Learning Systems*, 2023.

[32] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*, 2021.

[33] 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*, pages 24480–24489, 2023.

[34] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2337–2346, 2019.

[35] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.

[36] Andreas Rossler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner. Faceforensics++: Learning to detect manipulated facial images. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1–11, 2019.

[37] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115:211–252, 2015.

[38] Peize Sun, Yi Jiang, Shoufa Chen, Shilong Zhang, Bingyue Peng, Ping Luo, and Zehuan Yuan. Autoregressive model beats diffusion: Llama for scalable image generation. *arXiv preprint arXiv:2406.06525*, 2024.

[39] Chuangchuang Tan, Yao Zhao, Shikui Wei, Guanghua Gu, and Yunchao Wei. Learning on gradients: Generalized artifacts representation for gan-generated images detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12105–12114, 2023.

[40] 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*, pages 28130–28139, 2024.

[41] Ruixiang Tang, Dehan Kong, Longtao Huang, et al. Large language models can be lazy learners: Analyze shortcuts in in-context learning. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 4645–4657, 2023.

[42] Keyu Tian, Yi Jiang, Zehuan Yuan, Bingyue Peng, and Liwei Wang. Visual autoregressive modeling: Scalable image generation via next-scale prediction. *Advances in neural information processing systems*, 37:84839–84865, 2024.

[43] Vladimir Vapnik. *The nature of statistical learning theory*. Springer science & business media, 2013.

[44] Qizhou Wang, Yong Lin, Yongqiang Chen, Ludwig Schmidt, Bo Han, and Tong Zhang. A sober look at the robustness of clips to spurious features. *Advances in Neural Information Processing Systems*, 37:122484–122523, 2024.

[45] 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*, pages 8695–8704, 2020.

[46] 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*, pages 22445–22455, 2023.

[47] Chengyue Wu, Xiaokang Chen, Zhiyu Wu, Yiyang Ma, Xingchao Liu, Zizheng Pan, Wen Liu, Zhenda Xie, Xingkai Yu, Chong Ruan, et al. Janus: Decoupling visual encoding for unified multimodal understanding and generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 12966–12977, 2025.

[48] Jinheng Xie, Weijia Mao, Zechen Bai, David Junhao Zhang, Weihao Wang, Kevin Qinghong Lin, Yuchao Gu, Zhijie Chen, Zhenheng Yang, and Mike Zheng Shou. Show-o: One single transformer to unify multimodal understanding and generation. *arXiv preprint arXiv:2408.12528*, 2024.

[49] Zhuo Xu, Xiang Xiang, and Yifan Liang. Overcoming shortcut problem in vlm for robust out-of-distribution detection. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 15402–15412, 2025.

[50] Zhiyuan Yan, Jiangming Wang, Peng Jin, Ke-Yue Zhang, Chengchun Liu, Shen Chen, Taiping Yao, Shouhong Ding, Baoyuan Wu, and Li Yuan. Orthogonal subspace decomposition for generalizable ai-generated image detection. *arXiv preprint arXiv:2411.15633*, 2024.

[51] Fisher Yu, Ari Seff, Yinda Zhang, Shuran Song, Thomas Funkhouser, and Jianxiong Xiao. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint arXiv:1506.03365*, 2015.

[52] Haifeng Zhang, Qinghui He, Xiuli Bi, Weisheng Li, Bo Liu, and Bin Xiao. Towards universal ai-generated image detection by variational information bottleneck network. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 23828–23837, 2025.

[53] Haifeng Zhang, Qinghui He, Xiuli Bi, Bo Liu, Chi-Man Pun, and Bin Xiao. Combating pattern and content bias: Adversarial feature learning for generalized ai-generated image detection. *IEEE Transactions on Image Processing*, 2026. doi: 10.1109/TIP.2026.3742643.

[54] Laixin Zhang, Shuaibo Li, Wei Ma, and Hongbin Zha. Truemoe: Dual-routing mixture of discriminative experts for synthetic image detection. *arXiv preprint arXiv:2509.15741*, 2025.

[55] Ziyin Zhou, Yunpeng Luo, Yuanchen Wu, Ke Sun, Jiayi Ji, Ke Yan, Shouhong Ding, Xiaoshuai Sun, Yunsheng Wu, and Rongrong Ji. Aigi-holmes: Towards explainable and generalizable ai-generated image detection via multimodal large language models. *arXiv preprint arXiv:2507.02664*, 2025.

[56] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017.

[57] 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, 2024.

A Detailed Computations and Results of the Preliminaries

This section provides detailed computations and additional explanations for the empirical validation of the Generative Artifact Manifold Clustering (GAMC) hypothesis in Sec. 3.1. Appendix A.1 validates hypothesis (i), namely that different generators exhibit stable distribution shifts along a common set of artifact-related dimensions. Appendix A.2 validates hypothesis (ii), namely that these artifact-related dimensions are implicitly encoded in pre-trained multimodal representations. Appendix A.3 provides additional support for hypothesis (iii), namely that generator artifacts remain separable in deep representation spaces even when semantic content is aligned.

A.1 Statistical Analysis of Handcrafted Artifact Features

To validate hypothesis (i), we selected four representative generative models: Stable Diffusion v1.4, Midjourney, ADM, and BigGAN. For Stable Diffusion v1.4, Midjourney, and ADM, we generated 300 images using identical prompts. For BigGAN, which is class-conditional rather than prompt-driven, we generated images from semantically matched categories so that the content was aligned as closely as possible across generators and the observed differences mainly reflected generator-dependent artifacts rather than semantic variations. We extracted 25 handcrafted artifact-related statistical dimensions, covering frequency-domain energy and spectral slope, color statistics, brightness and contrast, edge density and strength, texture/GLCM features, compression traces, and noise statistics. These dimensions are commonly used to describe visual traces that may be introduced by generative processes [9, 29, 1, 16].

Z-score normalization. For each artifact dimension, we first computed the Z-score of the mean value of each generator:

$$Z_g^{(m)} = \frac{\bar{x}_g^{(m)} - \mu^{(m)}}{\sigma^{(m)}}, \tag{13}$$

where $\bar{x}_g^{(m)}$ denotes the mean value of generator g on artifact dimension m, and $\mu^{(m)}$ and $\sigma^{(m)}$ denote the mean and standard deviation across all generators on that dimension. The Z-score measures whether one generator has a higher or lower value than the overall generator average on a given artifact dimension. A larger absolute Z-score indicates a stronger generator-specific deviation.

One-way ANOVA. To examine whether the generator-dependent deviations are statistically significant, we conducted one-way analysis of variance (ANOVA) for each artifact dimension, using generator identity as the categorical factor. For a given artifact dimension, the null hypothesis is that all generators have the same mean value:

$$H_0 : \mu_1 = \mu_2 = \cdots = \mu_G, \tag{14}$$

where G is the number of generators. The ANOVA test statistic is

$$F = \frac{MS_{\text{between}}}{MS_{\text{within}}}, \tag{15}$$

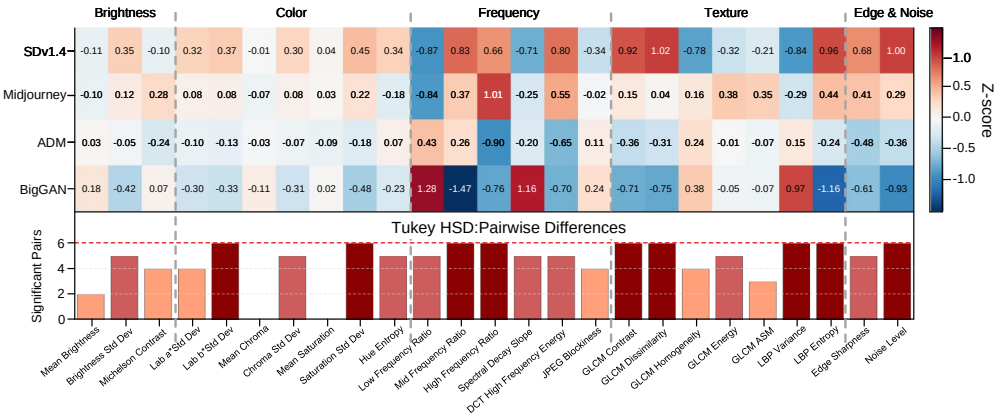


Figure 6: Z-score distributions of 25 handcrafted artifact-related statistics for images generated by Stable Diffusion v1.4, Midjourney, ADM, and BigGAN under identical prompts. Each value indicates how much the mean statistic of one generator deviates from the average across generators. The distinct deviation patterns show that different generators leave generator-dependent traces across multiple artifact dimensions.

where MS_{between} and MS_{within} are the between-generator and within-generator mean squares, respectively. The corresponding p-value is computed from the right tail of the F distribution:

$$p = P\left(F_{G-1,N-G} \geq F_{\text{obs}}\right), \tag{16}$$

where N is the total number of samples. A small p-value indicates that at least one generator has a significantly different mean on that artifact dimension.

Effect sizes. Statistical significance alone does not indicate the magnitude of the generator-dependent difference. Therefore, we also report two effect-size measures, η^2 and ω^2:

$$\eta^2 = \frac{SS_{\text{between}}}{SS_{\text{total}}}, \tag{17}$$

$$\omega^2 = \frac{SS_{\text{between}} - df_{\text{between}} MS_{\text{within}}}{SS_{\text{total}} + MS_{\text{within}}}. \tag{18}$$

Here, SS_{between} and SS_{total} denote the between-generator and total sums of squares, and $df_{\text{between}} = G-1$ is the between-generator degree of freedom. η^2 measures the fraction of total variance explained by generator identity. ω^2 is a bias-corrected effect-size estimate and is usually more conservative than η^2. In Fig. 2, a larger ω^2 means that generator identity explains a larger fraction of the variation along that artifact dimension.

Post-hoc pairwise comparison. After ANOVA, we further applied Tukey's Honestly Significant Difference (HSD) test for pairwise comparisons between generators. ANOVA only indicates whether a dimension differs across generators overall, while Tukey HSD identifies which specific generator pairs differ significantly. It controls the family-wise error rate across multiple pairwise comparisons, making it suitable for comparing all generator pairs on each artifact dimension.

Results. As shown in Fig. 6, the four generators show distinct deviation patterns across the 25 artifact-related dimensions. Stable Diffusion v1.4 exhibits pronounced deviations in low-frequency energy and brightness statistics, Midjourney shows consistent shifts in color and saturation statistics, ADM differs notably in texture and edge-strength dimensions, and BigGAN displays distinctive patterns in high-frequency energy and noise-related statistics. The one-way ANOVA results show that 23 out of 25 dimensions are statistically significant at $p < 0.01$, with a mean ω^2 of 0.2699. Tukey HSD comparisons further confirm that most generator pairs differ significantly, especially on frequency-, brightness-, and texture-related dimensions. These results support hypothesis (i): different generative models introduce stable distribution shifts along a common set of artifact-related dimensions.

A.2 Linear Probing of Artifact Statistics in CLIP Features

To validate hypothesis (ii), we examined whether artifact-related handcrafted statistics are implicitly encoded in CLIP representations. We froze the CLIP image encoder and trained a linear probe to predict handcrafted artifact statistics from CLIP image features. This setting follows the standard linear probing protocol: if a simple linear mapping can recover an artifact statistic from CLIP features, then the corresponding artifact-related information is at least partially encoded in the representation.

Let $X \in \mathbb{R}^{n\times d}$ denote the CLIP feature matrix and $Y \in \mathbb{R}^{n\times M}$ denote the handcrafted artifact-feature matrix. The linear probe is defined as

$$\hat{Y} = XW + b, \tag{19}$$

where W and b are learnable parameters. We optimize the mapping using ridge regression:

$$\min_{W,b} \|\hat{Y} - Y\|_2^2 + \lambda\|W\|_2^2, \tag{20}$$

where λ is the regularization coefficient. The ridge penalty limits overfitting and keeps the probe simple, so the result reflects whether the information is linearly recoverable rather than whether a high-capacity model can memorize the target statistics.

Evaluation metrics. For each handcrafted feature, we evaluated the prediction quality using three complementary metrics. First, the coefficient of determination R^2 measures the fraction of variance explained by the linear prediction:

$$R^2 = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \bar{y})^2}. \tag{21}$$

Second, Pearson correlation measures the linear association between predicted and ground-truth feature values. Third, Spearman correlation measures rank-level association and is more robust to monotonic nonlinear relationships and outliers. We also conducted significance tests for the correlations, where the null hypothesis is that the predicted and ground-truth values are uncorrelated.

Valid probed features. Although 25 handcrafted dimensions were extracted in Appendix A.1, the probing analysis used 23 valid handcrafted features after removing dimensions that were degenerate or not consistently defined across all samples. This avoids unstable correlation estimates caused by nearly constant or ill-defined target statistics.

Results. As shown in Fig. 3(a)–(c), the linear probe achieves non-trivial predictability for many artifact-related statistics. Across the 23 valid handcrafted features, the median absolute Pearson correlation is 0.551, and 20 out of 23 features are statistically significant at $p < 0.001$. Frequency-, texture-, color-, and brightness-related features are more predictable than compression- and noise-related features. This indicates that CLIP representations preserve artifact-related information beyond semantic content, supporting hypothesis (ii).

A.3 Separability of Generator Artifacts in Learned Representations

To provide additional support for hypothesis (iii), we trained a real–fake classifier on CLIP features and extracted the learned artifact representations before the final classification layer. We then visualized these representations using t-SNE for images generated by Stable Diffusion v1.4, Midjourney, ADM, and BigGAN under semantically aligned prompts.

As shown in Fig. 3(d), samples from different generators form clearly separated clusters in the learned representation space. This visualization suggests that generator-dependent artifact information remains distinguishable even when semantic content is aligned. Together with the handcrafted-feature analysis in Appendix A.1 and the linear-probing results in Appendix A.2, this provides empirical support for the GAMC hypothesis from three complementary perspectives: handcrafted artifact statistics differ across generators, these statistics are encoded in CLIP features, and generator-specific artifact representations remain separable in deep feature space.

Overall, Appendix A supports the motivation of SA-AFL. Since pre-trained representations encode both semantic information and diverse artifact-related cues, detectors may exploit spurious semantics

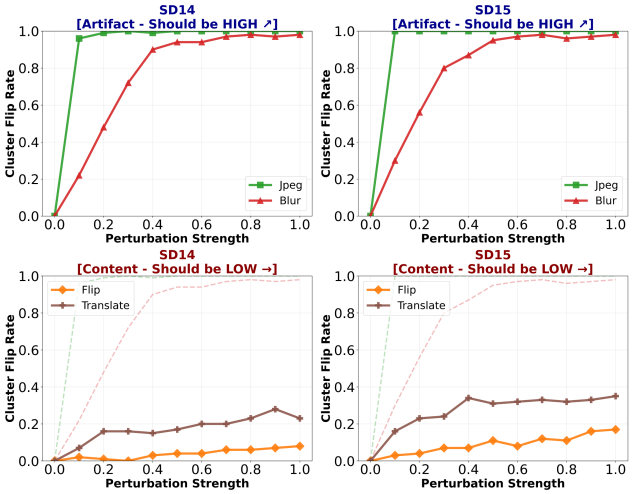


Figure 7: Controlled interventions for validating generative artifacts. Artifact-direction edits (JPEG compression, blur) progressively invert cluster assignments, whereas semantic-only permutations (spatial/geometric rearrangements preserving pixel-value distributions) have much weaker impact.

or over-rely on a few salient artifact dimensions when trained on limited generators. This motivates our goal of suppressing semantic shortcuts while preserving balanced, multi-dimensional artifact evidence for robust cross-generator generalization.

B Controlled Intervention Analysis of Artifact Subspaces

To verify that the principal components of these clusters correspond to generative artifacts, we perform controlled artifact/semantic interventions on them, as shown in Fig. 7. When applying controlled interventions along artifact directions (e.g., JPEG compression, blurring), the clusters invert as the intervention strength increases. In contrast, semantic-only permutations of the image—such as spatial or geometric rearrangements that leave pixel value distributions unchanged—have substantially weaker effects on the clusters. These results indicate that the clusters primarily represent generative artifact structures.

C Additional Experimental Results

This section provides additional experimental details and analyses that complement the main paper. The goal of these supplementary experiments is three-fold. First, we provide a more complete description of the experimental protocols and evaluation fairness. Second, we directly diagnose whether SA-AFL reduces shortcut reliance, rather than only improving final detection accuracy. Third, we study practical factors that may affect the use of SA-AFL, including the choice of cluster numbers, alternative clustering algorithms, visual backbones, computational overhead, and robustness under additional post-processing operations.

C.1 Detailed Experimental Setup and Evaluation Fairness

Training set. Following common practice, we use images generated by Stable Diffusion v1.4 [35] in the GenImage dataset, together with their corresponding real images, as our training data, resulting in a total of 32,000 samples (real + generated). In addition, the ForenSynths training set contains four categories, each with 18,000 real images from LSUN [51] and an equal number of ProGAN-generated images [18], a configuration widely adopted in early studies.

Testing set. To evaluate the generalization ability of our method across different generative models, we include images produced by seven advanced generators in the GenImage test set, along with their corresponding real images. These models include Stable Diffusion v1.4/1.5 [35], VQDM [12], ADM [7], GLIDE [32], Wukong, and Midjourney. We further use the ForenSynths test set, which contains images generated by 6 models, primarily from the GAN family [18, 56, 34, 6, 19]. The corresponding

real images are sourced from six widely used datasets: ImageNet [37], LSUN [51], FaceForensics++ [36], CelebA-HQ [18], CelebA [30], and COCO [28].

Cluster selection in the main experiments. The main experiments use data-driven cluster selection rather than manually tuned fixed cluster numbers. Given a candidate cluster number K, K-means is run on L_2-normalized source-domain features, and the Davies–Bouldin Index (DBI) is computed. The cluster number with the lowest DBI is selected:

$$K^\star = \arg\min_{K \in \mathcal{K}} \mathrm{DBI}\big(\{\bar{x}_i\}, \ell^K\big), \tag{22}$$

where $\bar{x}_i = x_i / \|x_i\|_2$ denotes the normalized feature, ℓ^K denotes the K-means assignment under cluster number K, and $\mathcal{K}$ is the candidate set. A lower DBI indicates more compact and better separated clusters. This strategy is applied to both Stage-I semantic clustering and Stage-II artifact clustering. The fixed-K experiments in Appendix C.4 are sensitivity studies and should not be confused with the DBI-guided configuration used for the main results.

Fair comparison with baselines. Except for UnivFD in Protocol 3, all competing methods are re-trained or re-evaluated using their official implementations under the same train/test protocols and the same test sets. For methods that require thresholding, we use a consistent decision threshold of 0.5 unless otherwise specified by the original method. The only exception is UnivFD in Protocol 3, where we report the official benchmark result. We explicitly mark this exception to avoid mixing official benchmark numbers with re-trained results.

Evaluation metrics. We report accuracy (ACC) and average precision (AP). ACC measures the binary real–fake classification performance under a fixed threshold, while AP evaluates ranking quality across thresholds. For tables that summarize results over multiple generators, we report the mean value across the corresponding test generators. Unless otherwise stated, all additional analyses in this section are conducted under Protocol 1, since it contains both diffusion-to-diffusion and diffusion-to-GAN generalization and is therefore suitable for diagnosing shortcut reliance.

C.2 Empirical Analysis of Shortcut Suppression

The main paper shows that SA-AFL improves cross-generator generalization. We further conduct an empirical analysis from the perspective of the feature space to examine how the proposed method suppresses shortcut reliance. We focus on two types of shortcuts: semantic shortcuts and generator-specific artifact shortcuts. A semantic shortcut means that the detector exploits semantic content that happens to correlate with the real–fake label in the source domain. A generator-specific shortcut means that the detector relies on artifact cues that are highly predictive for a particular source generator but are not sufficiently shared across unseen generators.

Linear probes for semantic and generator-specific information. We first use linear probes to quantify how much shortcut information is recoverable from the final representation. Let r_i denote the final feature representation before the binary classifier. We freeze the trained detector and train a linear classifier on r_i to predict the pseudo-semantic cluster label s_i. If the representation still contains strong semantic information, the linear probe should recover s_i with high accuracy. Thus, lower semantic-probe accuracy indicates weaker semantic shortcut information in the final representation.

Similarly, we train another linear probe to predict the generator identity. If the learned representation is strongly organized by generator-specific artifacts, the generator ID should be easy to recover. In contrast, if the detector learns a more shared fake-discriminative representation, generator identity should become less recoverable. Therefore, lower generator-probe accuracy suggests reduced dependence on generator-specific shortcuts.

Artifact-cluster compactness and balance. In addition to probing semantic and generator labels, we analyze the structure of artifact clusters after training. Let $\mathcal{C}_k$ denote the set of samples assigned to artifact cluster k, and let μ_k be the cluster center. We compute three statistics:

$$D_{\mathrm{inter}} = \frac{2}{K_a(K_a-1)} \sum_{p<q} \|\mu_p - \mu_q\|_2, \tag{23}$$

Table 7: Shortcut diagnosis on the final representation. Lower semantic-probe and generator-probe accuracies indicate that semantic and generator-specific shortcut information is less recoverable. Lower cluster-structure statistics indicate that artifact modes become more compact and balanced after self-adversarial learning.

Method	Semantic Probe ↓	Generator Probe ↓	Inter-cluster Dist. ↓	Intra-cluster Var. ↓	Intra-dispersion Var. ↓
Baseline	0.909	0.735	10.89	571.47	4945.72
SA-AFL	0.625	0.584	4.65	33.21	625.52

Table 8: Controlled semantic shortcut test. Person and car are shared-semantic control classes. Real dog and fake bird are shortcut-conflicting categories that reverse the semantic-label correlation introduced during training.

Method	Real person	Fake person	Real car	Fake car	Real dog	Fake bird	Shortcut error ↓
Baseline	90.1	90.0	89.9	94.7	68.3	71.0	30.4
SA-AFL	98.4	93.2	94.6	99.3	85.6	89.9	12.3

$$V_{\text{intra}} = \frac{1}{K_a} \sum_{k=1}^{K_a} \frac{1}{|\mathcal{C}_k|} \sum_{i \in \mathcal{C}_k} \|r_i - \mu_k\|_2^2, \tag{24}$$

$$V_{\text{disp}} = \text{Var}\left(\left\{ \frac{1}{|\mathcal{C}_k|} \sum_{i \in \mathcal{C}_k} \|r_i - \mu_k\|_2^2 \right\}_{k=1}^{K_a} \right). \tag{25}$$

Here, D_{inter} measures the average distance between artifact-cluster centers, V_{intra} measures the average within-cluster variance, and V_{disp} measures how imbalanced the within-cluster variances are across clusters. Before self-adversarial learning, artifact clusters can be highly separated and imbalanced, meaning that a few salient artifact modes may dominate the decision boundary. After self-adversarial learning, we expect the representation to become more compact and balanced, forcing the detector to integrate evidence from multiple artifact dimensions.

As shown in Table 7, SA-AFL reduces semantic-probe accuracy from 0.909 to 0.625. This indicates that semantic cluster labels become substantially harder to recover from the final representation. SA-AFL also reduces generator-probe accuracy from 0.735 to 0.584, suggesting that the final representation is less organized around generator identity. Meanwhile, the artifact clusters become much more compact and balanced: the average inter-cluster distance decreases from 10.89 to 4.65, the average intra-cluster variance decreases from 571.47 to 33.21, and the dispersion of intra-cluster variances decreases from 4945.72 to 625.52. These results support the intended behavior of SA-AFL: the model suppresses semantic and generator-specific shortcut information while forming a more shared fake-discriminative representation.

Controlled semantic shortcut test. We further construct a controlled test that directly injects semantic-label correlation during training. In this test, the category dog appears only in fake training images, while the category bird appears only in real training images. Meanwhile, person and car appear in both real and fake training images and serve as shared-semantic control classes. Therefore, a detector that learns semantic shortcuts should treat dog as a cue for fake and bird as a cue for real. At test time, we introduce shortcut-conflicting samples from the same generator: real dog and fake bird. If the model relies on the injected semantic shortcut, it should tend to misclassify real dog as fake and fake bird as real.

We define the shortcut error as the average error rate on the two shortcut-conflicting categories:

$$\text{ShortcutError} = \frac{(1 - \text{Acc}_{\text{real dog}}) + (1 - \text{Acc}_{\text{fake bird}})}{2}. \tag{26}$$

The control classes person and car are used to verify that the detector is not simply failing on the entire test set. A meaningful semantic-shortcut failure should mainly appear on real dog and fake bird, while the shared-semantic control classes remain relatively normal.

Table 8 shows that the baseline performs well on the shared-semantic control classes, achieving 90.0–94.7% accuracy on person and car. However, it drops sharply on the shortcut-conflicting categories, reaching only 68.3% on real dog and 71.0% on fake bird. This indicates that the baseline

Table 9: Effect of semantic–artifact orthogonality. The orthogonality constraint reduces semantic leakage while preserving conditional artifact predictability and increasing relative sensitivity to artifact changes.

Model	Semantic Leakage Probe ↓	Conditional Artifact Probe ↑	Δ_{sem} ↓	Δ_{art}	$\Delta_{\mathrm{art}}/\Delta_{\mathrm{sem}}$ ↑
w/o orthogonality	0.600	0.830	0.336	0.366	1.090
w/ orthogonality	0.533	0.847	0.207	0.266	1.281

has indeed learned the injected semantic-label correlation. In contrast, SA-AFL maintains strong performance on the control classes and substantially improves the shortcut-conflicting cases to 85.6% and 89.9%, respectively. The shortcut error decreases from 30.4% to 12.3%. Since the large errors of the baseline are concentrated on the semantically biased categories rather than all categories, this test provides direct evidence that SA-AFL mitigates semantic shortcut reliance.

C.3 Effect of Semantic–Artifact Decoupling

SA-AFL introduces semantic–artifact decoupling regularization to reduce redundant overlap between the semantic and artifact branches. This regularization is not intended to remove all information from the artifact branch. Instead, it should suppress semantic leakage while preserving useful artifact-related variation. We therefore conduct an orthogonality-specific analysis to verify this behavior.

Semantic leakage probe. We train a linear probe on the artifact branch to predict semantic cluster labels. If the artifact branch still contains strong semantic information, this probe should achieve high accuracy. A lower semantic leakage probe accuracy therefore indicates that semantic content is less recoverable from the artifact branch.

Conditional artifact probe. We also evaluate whether artifact information remains usable after applying the orthogonality constraint. For this purpose, we train a conditional artifact probe to predict artifact clusters while controlling for semantic clusters. If the orthogonality constraint simply removes useful artifact information, the conditional artifact probe should drop substantially. If it mainly removes semantic leakage, the conditional artifact probe should be preserved or improved.

Intervention sensitivity. Finally, we conduct an intervention sensitivity analysis. Following the controlled intervention analysis in Appendix B, we consider semantic-only interventions and artifact-only interventions. Semantic-only interventions change semantic layout or spatial/geometric arrangement while leaving pixel-value distributions largely unchanged. Artifact-only interventions change artifact-related image statistics, such as compression or blur. Let $z_a(x)$ denote the artifact-branch representation. We define the average sensitivity to semantic-only changes as

$$\Delta_{\mathrm{sem}} = \mathbb{E}_x \left[\| z_a(T_{\mathrm{sem}}(x)) - z_a(x) \|_2 \right], \tag{27}$$

and the average sensitivity to artifact-only changes as

$$\Delta_{\mathrm{art}} = \mathbb{E}_x \left[\| z_a(T_{\mathrm{art}}(x)) - z_a(x) \|_2 \right]. \tag{28}$$

We further compute the relative artifact sensitivity

$$R = \frac{\Delta_{\mathrm{art}}}{\Delta_{\mathrm{sem}}}. \tag{29}$$

A higher R indicates that the artifact branch is relatively more responsive to artifact changes than to semantic changes.

As shown in Table 9, adding the orthogonality constraint reduces semantic leakage probe accuracy from 0.600 to 0.533. This suggests that semantic information becomes less recoverable from the artifact branch. At the same time, the conditional artifact probe slightly improves from 0.830 to 0.847, indicating that useful artifact information is not removed. The intervention analysis further supports this conclusion. The artifact branch becomes less sensitive to semantic-only changes, with Δ_{sem} decreasing from 0.336 to 0.207. Although Δ_{art} also decreases, the relative sensitivity ratio increases from 1.090 to 1.281. This means that, after orthogonality regularization, the artifact branch responds more selectively to artifact variation relative to semantic variation. Overall, the orthogonality term reduces semantic–artifact redundancy rather than suppressing valid generative artifacts.

Table 10: Sensitivity to K_s and K_a under Protocol 1. Left: varying K_s with $K_a = 8$. Right: varying K_a with $K_s = 100$. These are fixed-K diagnostic experiments; the main results use DBI-guided automatic cluster selection.

$K_a = 8$			$K_s = 100$		
K_s	ACC	AP	K_a	ACC	AP
10	90.15	96.91	2	89.99	98.43
25	90.02	98.50	4	86.09	97.68
50	91.07	98.62	6	89.04	98.33
75	91.24	97.77	8	91.38	97.73
100	91.38	97.73	10	91.68	98.55
125	91.77	98.33	12	92.95	98.54
150	89.10	97.84	14	90.73	98.55
175	92.04	98.15	16	89.36	98.16
200	92.17	95.64	18	91.41	98.82
–	–	–	20	92.87	98.61

Table 11: Sensitivity to K_s and K_a under Protocol 2. Left: varying K_s with $K_a = 8$. Right: varying K_a with $K_s = 4$. Because the ProGAN training set contains only four semantic categories, the reasonable range of K_s is smaller than in Protocol 1.

$K_a = 8$			$K_s = 4$		
K_s	ACC	AP	K_a	ACC	AP
2	86.84	94.05	4	87.95	95.38
3	86.27	93.84	6	88.18	95.56
4	87.84	95.29	8	87.84	95.29
5	85.38	94.08	10	87.94	95.36
6	87.00	95.37	12	88.43	95.41
–	–	–	14	88.41	95.69

C.4 Sensitivity to Semantic and Artifact Cluster Numbers

SA-AFL uses semantic and artifact clusters as pseudo-labels for adversarial learning. A natural question is whether its performance depends on a fragile manually chosen pair of cluster numbers. We therefore conduct sensitivity studies on K_s and K_a under Protocol 1 and Protocol 2. These experiments use fixed cluster numbers for analysis purposes, while the main results use data-driven DBI-guided cluster selection.

Interpretation of K_s and K_a. The two cluster numbers have different meanings. The semantic cluster number K_s controls the granularity of pseudo-semantic partitions. If K_s is too small, semantically different contents may be merged, causing weak semantic supervision. If K_s is too large, visually similar semantic modes may be split into sparse and unstable pseudo-labels. The artifact cluster number K_a controls the granularity of artifact modes among generated images. If K_a is too small, distinct artifact modes may be merged. If K_a is too large, pseudo-labels may become noisy and overly fragmented. Therefore, we expect a reasonable range rather than a single universally optimal value.

Table 10 shows that SA-AFL remains strong across a wide range of K_s and K_a under Protocol 1. When $K_a = 8$, AP remains above 95.6% for all tested K_s values, and ACC remains mostly above 90%. The performance does not collapse when K_s varies from 10 to 200, indicating that the method does not rely on a single fragile semantic-clustering choice. When $K_s = 100$, SA-AFL also remains effective over a broad range of K_a. Very small K_a values can be suboptimal because they merge different artifact modes, while medium-to-large values generally preserve strong performance. These results suggest that artifact self-adversarial learning benefits from sufficiently fine artifact partitions, but does not require exact manual tuning.

Table 11 shows a similar trend under Protocol 2. Because the ProGAN source training set contains only four semantic categories, the useful range of K_s is naturally smaller. Within this range, SA-AFL remains stable. For artifact clusters, $K_a = 4$ to 14 all produce similar results, showing that the method is not sensitive to a narrow artifact-cluster choice. Overall, Tables 10 and 11 indicate that K_s

Table 12: Alternative clustering modules under Protocol 1. The K-means row is a fixed-K controlled baseline in this clustering-comparison setting, not the DBI-guided configuration used for the main results.

Clustering module	Diffusion Models		GANs		All	
	ACC	AP	ACC	AP	ACC	AP
Spectral Clustering	93.44	99.20	89.66	94.88	92.70	97.67
GMM	92.57	97.82	88.50	94.65	90.69	96.36
UMAP + K-means	95.95	98.32	91.56	95.26	93.92	96.91
K-means	92.90	97.76	89.28	95.03	91.23	96.50

Table 13: Automatic K selection under Protocol 1. The selected K_s and K_a are determined using source-domain training features only.

Criterion	ACC	AP	Selected K
Silhouette Score	90.34	96.35	$(K_s = 137,\ K_a = 6)$
Calinski–Harabasz	89.68	98.25	$(K_s = 20,\ K_a = 2)$
Davies–Bouldin	92.56	97.73	$(K_s = 128,\ K_a = 6)$

mainly reflects the semantic diversity of the source training set, while K_a reflects the granularity of artifact modes. SA-AFL is robust across reasonable ranges of both parameters.

C.5 Alternative Clustering, Automatic K Selection, and K-free Clustering

The clustering step in SA-AFL is used to construct pseudo-labels for adversarial shortcut mitigation. The main contribution is therefore not K-means itself, but the two-stage mechanism that uses semantic and artifact pseudo-labels to suppress shortcut reliance. To verify this, we conduct three additional analyses. First, we replace K-means with alternative clustering or manifold-learning modules. Second, we use standard unsupervised criteria to select K_s and K_a automatically. Third, we test clustering algorithms that do not require pre-specifying K.

Alternative clustering modules. We compare four clustering modules under the same SA-AFL training framework. K-means is the fixed-K reference variant. Spectral Clustering captures local graph geometry through a neighborhood graph. Gaussian Mixture Model (GMM) allows soft cluster assignments and anisotropic cluster structures. UMAP+K-means first maps features into a nonlinear low-dimensional manifold representation and then applies K-means. This last variant is particularly relevant to the GAMC hypothesis, because it can better capture continuous manifold-like structures before pseudo-label construction.

As shown in Table 12, SA-AFL remains effective under all clustering variants. This indicates that the effectiveness of the framework does not come from a particular K-means implementation. Among the tested alternatives, UMAP+K-means achieves the best overall ACC, improving the fixed-K K-means reference from 91.23% to 93.92%. This suggests that manifold-aware pseudo-labeling can further improve shortcut mitigation by producing more meaningful semantic or artifact partitions. These results are consistent with the GAMC hypothesis: generative artifacts are not necessarily organized as isolated Euclidean blobs, but may follow nonlinear manifold-like structures in deep representation space.

Automatic K selection. We next evaluate whether standard data-driven criteria can automatically select K_s and K_a using only source-domain training features. We consider three widely used unsupervised criteria: Silhouette Score, Calinski–Harabasz Index, and Davies–Bouldin Index. All three methods use K-means as the underlying clustering algorithm, but the cluster numbers are selected automatically from candidate ranges. This setting tests whether manual prior analysis of K_s and K_a is necessary.

Table 13 shows that automatic K selection remains competitive. All three criteria achieve ACC above 89% and AP above 96%. In practice, SA-AFL can use source-domain features to select cluster numbers automatically.

Table 14: K-free clustering under Protocol 1. These methods automatically determine the number of clusters and do not require manually specifying K_s or K_a.

Method	ACC	AP
HDBSCAN	90.45	97.50
X-means	90.73	98.85
G-means	89.02	97.85

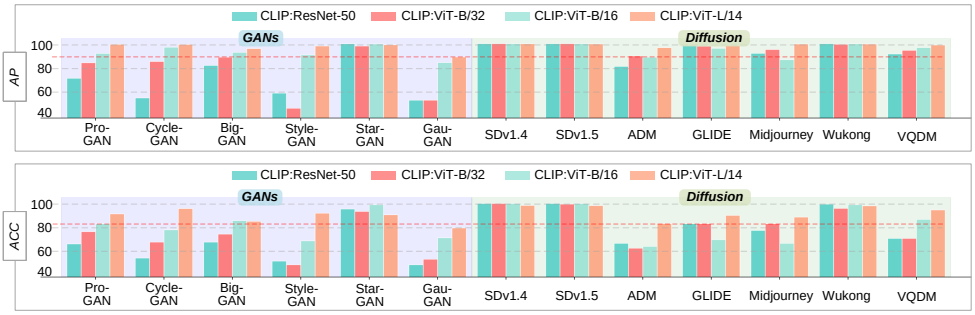


Figure 8: Performance of SA-AFL variants with different CLIP backbones under Protocol 1. CLIP ViT-L/14 yields the strongest results, while other CLIP backbones also remain effective.

Table 15: Transferability of SA-AFL across different visual encoders under Protocol 1. For each backbone, the baseline and SA-AFL use the same encoder, and the gain is computed in ACC points.

Backbone	Baseline ACC	SA-AFL ACC	Gain
CLIP ViT-B/16	80.98	85.39	+4.41
MetaCLIP	78.03	83.29	+5.26
DINOv2	79.99	83.27	+3.28
CLIP ViT-L/14	84.55	92.56	+8.01

K-free clustering. We further test clustering algorithms that do not require pre-specifying the number of clusters. We consider HDBSCAN, X-means, and G-means. HDBSCAN estimates clusters based on density and can treat low-density samples as noise. X-means extends K-means by automatically growing the number of clusters according to a model-selection criterion. G-means tests whether each cluster follows a Gaussian distribution and recursively splits clusters when necessary. These methods evaluate whether SA-AFL can remain effective even when no explicit K_s or K_a is specified.

As shown in Table 14, SA-AFL remains strong even with K-free clustering. X-means achieves the highest AP among the three methods, while HDBSCAN and G-means also produce competitive results. Together, Tables 10–14 show that SA-AFL is not dependent on a fragile, manually selected single pair of cluster numbers. A more accurate description is that SA-AFL is robust to reasonable clustering choices and can be implemented with either automatic K selection or K-free clustering.

C.6 Transferability Across Visual Backbones

The main experiments use CLIP ViT-L/14 because it provides strong general-purpose visual representations. Here, we study whether the effectiveness of SA-AFL depends on this specific backbone. We conduct two complementary analyses. First, we evaluate SA-AFL variants with different CLIP backbones, including CLIP ResNet-50, CLIP ViT-B/32, CLIP ViT-B/16, and CLIP ViT-L/14. Second, we compare the baseline detector and SA-AFL under multiple visual encoders, including CLIP ViT-B/16, MetaCLIP, DINOv2, and CLIP ViT-L/14.

As shown in Fig. 8, SA-AFL remains effective across different CLIP backbone configurations. CLIP ViT-L/14 achieves the strongest overall performance, likely because its larger capacity and attention-based global representation can capture richer artifact-sensitive cues. Table 15 further shows that the shortcut-mitigation framework consistently improves over the corresponding baseline across different visual encoders. The gains range from +3.28 to +8.01 ACC points. These results indicate

Table 16: Computational overhead comparison. Total parameters include the visual encoder and task-specific modules. Trainable parameters count only parameters updated during detector training. Inference time is measured per image.

Method	Total Params (M)	Trainable Params	GFLOPs	Infer. (ms)	Train/Epoch (s)	Mem. (MB)
UnivFD	303.9	769	77.8	10.9	8748.6	1268
CLIPping	389.7	24576	90.9	45.4	9889.6	1662
VIB-Net	306.3	2362113	77.8	11.3	7248.7	1278
SA-AFL	304.1	950500	77.8	12.3	9173.8	1354

Table 17: Extended robustness evaluation under Protocol 1. We report ACC under resizing, Gaussian noise, color perturbations, and a multi-step JPEG-to-blur pipeline.

Method	Scale			Gaussian Noise			Color Perturbation			JPEG→Blur
	$0.25\times$	$0.5\times$	$0.75\times$	$\sigma = 5$	$\sigma = 10$	$\sigma = 20$	Bright. 0.4	Contrast 0.4	Hue 0.1	
UnivFD	65.91	68.14	70.20	58.67	58.29	58.33	63.63	63.17	63.92	63.12
Effort	63.27	66.14	70.73	72.14	69.36	64.14	73.82	73.64	74.82	66.59
VIB-Net	62.04	64.85	65.81	71.54	69.92	64.85	72.27	71.96	73.12	64.65
SA-AFL	70.19	76.42	78.69	78.92	75.15	68.46	86.04	86.00	87.54	70.23

that SA-AFL is not tied to a single backbone. At the same time, stronger backbones can provide a better starting representation for identifying subtle and diverse artifact patterns.

C.7 Computational Overhead

SA-AFL introduces additional training-time components, including semantic and artifact clustering, adversarial discriminators, and decoupling regularization. We therefore report computational overhead and compare it with representative CLIP-based detectors. The additional clustering cost is incurred only during training when pseudo-labels are generated. During inference, clustering modules and adversarial heads are not required. The final inference path only uses the feature extractor and the real–fake classifier, so the inference overhead remains small.

Clustering cost. In our implementation, the Stage-I semantic clustering step takes 17.41 seconds, and the Stage-II artifact clustering step takes 3.12 seconds. These costs are small compared with the total per-epoch training time. Moreover, clustering is performed on extracted source-domain training features and does not scale with the number of test images. Therefore, SA-AFL does not introduce additional clustering computation at deployment time.

As shown in Table 16, SA-AFL has similar total parameter count and GFLOPs to other CLIP-based methods. Its inference latency is 12.3 ms per image, close to UnivFD and VIB-Net, and substantially lower than CLIPping. Although SA-AFL has more trainable parameters than linear-probing methods such as UnivFD, it has fewer trainable parameters than VIB-Net. The memory usage is also close to other CLIP-based detectors. Overall, the extra computation mainly affects training, while inference remains lightweight. This makes SA-AFL practical for deployment scenarios where the detector is trained offline and then applied to large numbers of test images.

C.8 Extended Robustness Evaluation

The main paper evaluates robustness to JPEG compression and Gaussian blur. Here, we extend the robustness analysis to additional post-processing operations that commonly occur during image transmission, editing, and online sharing. We consider four categories of perturbations: resizing, additive Gaussian noise, color perturbation, and multi-step post-processing. For resizing, we test scaling factors of $0.25\times$, $0.5\times$, and $0.75\times$. For Gaussian noise, we test noise levels $\sigma = 5$, $\sigma = 10$, and $\sigma = 20$. For color perturbation, we test brightness, contrast, and hue shifts. For multi-step post-processing, we test a JPEG-to-blur pipeline. All methods are evaluated under the same perturbed test images.

Table 17 shows that SA-AFL outperforms competing methods across all 10 additional perturbation settings. Under resizing, SA-AFL achieves the best accuracy at all three scales. Under Gaussian noise, SA-AFL also remains consistently better than UnivFD, Effort, and VIB-Net. The advantage is particularly large under color perturbations. Compared with the strongest baseline in each setting,

SA-AFL improves ACC by 12.22 points under brightness perturbation, 12.36 points under contrast perturbation, and 12.72 points under hue perturbation. Under the multi-step JPEG-to-blur pipeline, SA-AFL also achieves the highest accuracy. These results indicate that the robustness gain of SA-AFL is not limited to JPEG compression or Gaussian blur. Instead, reducing shortcut reliance helps the detector remain more stable under a wider range of realistic post-processing operations.

C.9 Discussion of Failure Cases

SA-AFL mitigates semantic shortcuts and dominant artifact shortcuts, but it still relies on the presence of transferable generative traces. We observe two representative challenging cases.

Low-artifact generated images. When recent high-fidelity generators produce images whose low-level statistics and texture patterns are very close to real images, the remaining artifact evidence may be weak, making detection more difficult.

Severe post-processing. Strong compression, noise, repeated resizing, denoising, or multi-step editing can attenuate generative traces. Although SA-AFL remains more robust than competing methods under the post-processing settings in Appendix C.8, extremely severe transformations may still reduce real–fake separability.

Future directions. Future work could improve SA-AFL in several directions. First, more realistic simulation of image propagation chains may help the detector learn artifact evidence that survives online sharing and editing. Second, incorporating additional source generators or synthetic augmentations may improve coverage of artifact modes. Third, combining shortcut-mitigated representation learning with uncertainty estimation could help identify samples whose artifact evidence is too weak for confident classification. Finally, more expressive manifold-aware clustering methods may further improve pseudo-label quality, as suggested by the UMAP+K-means results in Appendix C.5.

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 main claims in the abstract and Sec. 1 accurately reflect the paper's contributions and scope. The claims are supported by the GAMC analysis in Secs. 3.1 and 3.2, the SA-AFL framework in Sec. 4, and the cross-generator evaluations in Sec. 5.2.

   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: The paper discusses limitations and failure cases in Appendix C.9, including low-artifact generated images and severe post-processing. It also discusses robustness and computational overhead in Sec. 5.4, Appendix C.8, and Appendix C.7.

   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: [N/A]

Justification: The paper does not present formal theoretical results, theorems, or proofs. The GAMC hypothesis is supported empirically through handcrafted artifact analysis, linear probing, and representation visualization in Sec. 3.2 and Appendix A, rather than by formal proof.

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: The paper provides the information needed to reproduce the main experimental results. The method is specified in Sec. 4, while Sec. 5.1 and Appendix C.1 describe the datasets, train/test protocols, evaluation metrics, optimizer, hyperparameters, backbone, clustering procedure, and baseline evaluation settings.

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: [Yes]

   Justification: Code and model weights are publicly available at `https://github.com/oceanzhf/SA-AFL`. The experiments use publicly available datasets cited in the paper, and the implementation and evaluation settings are detailed in Sec. 5.1 and Appendix C.1.

   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: The paper specifies the training and test protocols, datasets, optimizer, learning rate, weight decay, batch size, backbone, clustering candidate ranges, adversarial schedule, and loss coefficients in Sec. 5.1. Additional details on data splits, evaluation fairness, automatic cluster selection, and metrics are provided in Appendix C.1.

   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: The paper reports statistical significance for the GAMC analysis, including ANOVA tests, effect sizes, and correlation significance in Sec. 3.2 and Appendix A.

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: Sec. 5.1 specifies the implementation environment, including PyTorch 2.7, CLIP ViT-L/14, batch size, and an NVIDIA RTX A6000 GPU. Appendix C.7 further reports clustering time, total and trainable parameters, GFLOPs, inference latency, training time per epoch, and memory usage.

   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: The research is intended for defensive AI-generated image detection and uses cited public datasets and models for evaluation. The paper does not introduce new human-subject data collection or release a high-risk generative system, and it documents limitations and possible failure cases in Appendix C.9.

   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: The paper discusses the positive societal motivation of reliable AIGI detection in Sec. 1, including mitigating misinformation and improving trust in digital media. Potential negative impacts include false positives that may incorrectly challenge authentic media, false negatives that may overstate trust in generated media, and misuse in over-automated moderation, therefore, deployment should be accompanied by human review, transparency, and careful threshold calibration.

    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: [N/A]

    Justification: The paper proposes a detector rather than a high-risk generative model, scraped dataset, or other asset that directly enables generation of harmful content. Existing public datasets are used for research evaluation, and no new high-risk data or generative model is released by this work.

    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: [No]

    Justification: The paper cites the original works for the datasets, pretrained encoders, generative models, and baseline methods used in the experiments. However, the current manuscript does not exhaustively list the license names and terms of use for every existing asset, so we answer No for full transparency.

    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: [N/A]

    Justification: The paper introduces a new training framework and experimental analysis, but it does not release a new dataset, benchmark, pretrained model, or other standalone asset in the current submission. Therefore, asset documentation requirements are not applicable.

    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: The paper does not involve crowdsourcing experiments or new research with human subjects. All experiments are conducted on existing image datasets and generated-image benchmarks.

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: The paper does not involve crowdsourcing or new human-subject studies, and no new data are collected from human participants. Therefore, IRB approval or equivalent human-subject review is not applicable.

    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: LLMs are not used as an important, original, or non-standard component of the core methodology. The proposed SA-AFL framework is based on visual feature learning, clustering, adversarial training, and real–fake image classification, any use of LLMs for writing, editing, or formatting, if any, does not affect the scientific methodology or originality of the work.

    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.