LEARNING ADAPTIVE EVIDENCE SELECTION FOR GENERALIZABLE AI-GENERATED IMAGE DETECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Generalizable AI-generated image (AIGI) detection aims to identify synthetic images and generalize to unseen generators. To improve the generalization ability, recent detectors increasingly adopt Vision Foundation Models (VFMs). However, most VFM-based detectors use the VFM as the image representation, leaving two issues unresolved. First, the global token is dominated by high-level semantic content, which can induce reliance on semantic content rather than generation-related forensic cues. Second, the global token focuses on a few patches, leaving complementary evidence in other patches underused. To address these issues, we propose the Adaptive Evidence Selection (**AES**) framework for generalizable AIGI detection. To reduce the influence of semantic content, we propose Cluster-centered Bilinear Aggregation (CBA), which suppresses shared semantic content and models local features within each cluster. To exploit forensic cues distributed across regions, we propose Determinantal Evidence Selection (DES), which selects cluster features by jointly evaluating task relevance and feature diversity. Experiments on four benchmarks demonstrate the effectiveness of our proposed method. Our source code will be released soon.

1 INTRODUCTION

As image generators produce increasingly realistic images that can deceive human observers, AI-generated image (AIGI) detection has become essential. In practical applications, detectors must identify images generated by unseen models, making out-of-domain generalization a central challenge. Early detectors often capture generator-specific artifacts and therefore generalize poorly to unseen generators (Wang et al., 2020; Tan et al., 2024b). Recently, Vision Foundation Models (VFMs), such as CLIP (Radford et al., 2021), have provided transferable visual representations and have been widely used for generalizable AIGI detection (Ojha et al., 2023; Koutlis & Papadopoulos, 2024; Liu et al., 2024).

Most VFM-based detectors use the global token as the image-level representation. However, it leads to two issues, as illustrated in Fig. 1(a). First, VFMs are encouraged to represent high-level image semantics. Consequently, the global token summarizes content not related to the image generation process. This reliance can hurt generalization when the image content or generator changes (Yang et al., 2026a). Second, the representation is spatially concentrated: a small number of salient patches can dominate the final prediction, while weak but complementary forensic cues in other regions receive little attention. This bias makes the decision depend on a limited set of image patches and leaves distributed evidence underused (Yang et al., 2026b).

These two issues suggest that a generalizable detector should meet two requirements. First, it should separate forensic features from the semantic content of the image. Second, it should collect these features from different image regions, rather than rely on only a few salient patches.

To address these issues, we propose Adaptive Evidence Selection (**AES**), a framework that separates forensic features from semantic content and selects the most relevant evidence. As shown in Fig. 2, AES consists of two components: Cluster-centered Bilinear Aggregation (CBA) for residual feature extraction and Determinantal Evidence Selection (DES) for evidence selection.

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

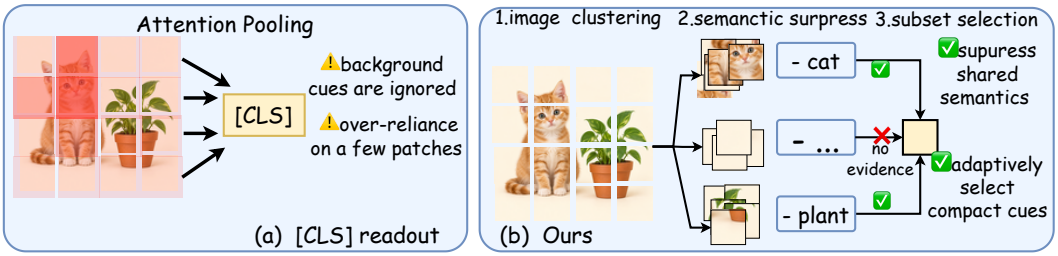

Figure 1: **Our motivation.** (a) The global token mainly focuses on a few salient patches and retains irrelevant semantic content. (b) Our method groups semantic patches , suppresses shared content, and selects compact evidence.

To separate forensic features from semantic content, we observe that an object or background entity typically spans multiple patches that share similar visual semantics. CBA partitions these patches into semantic clusters and subtracts each cluster centroid to eliminate local semantic baselines. To prevent downstream classifiers from linearly reconstructing the subtracted semantics, CBA further models second-order bilinear interactions across centered residuals, thoroughly suppressing semantic shortcuts while extracting subtle forensic evidence.

To collect forensic features from different image regions, DES uses class prototypes to score the relevance of each cluster-level feature. It then selects a compact subset that balances class relevance and feature diversity, combining evidence from the selected clusters, reducing reliance on salient patches while preserving complementary forensic cues. Experiments on four benchmarks demonstrate that our method improves both detection accuracy and generalization.

Our main contributions are summarized as follows:

- We propose Adaptive Evidence Selection (AES), a framework for generalizable AIGI detection. It reduces both the influence of semantic content and the reliance on salient regions.
- We propose Cluster-centered Bilinear Aggregation (CBA) to suppress semantics at the region level and extract a semantic-agnostic representation.
- We propose Determinantal Evidence Selection (DES) to select class-relevant and diverse evidence.
- Extensive experiments on four benchmarks demonstrate the effectiveness and generalization ability of our method across unseen generators.

2 Related Work

2.1 Generalizable AI-Generated Image Detection with VFMs

Early CNN-based detectors, such as CNNSpot (Wang et al., 2020), NPR (Tan et al., 2024b), and FreqNet (Tan et al., 2024a), can capture generator-specific artifacts but struggle to generalize to unseen generators. Recently, many methods adopt VFMs such as CLIP (Radford et al., 2021) to exploit their generalization ability. VFM-based methods broadly follow two adaptation paradigms: feature enhancement and parameter adaptation. Feature enhancement methods aim to extract richer representations from VFMs. For instance, UnivFD (Ojha et al., 2023) linearly probes CLIP features, RINE (Koutlis & Papadopoulos, 2024) aggregates multi-layer representations, and FatFormer (Liu et al., 2024) aligns visual and frequency cues. Parameter adaptation methods fine-tune a VFM with only a small number of trainable parameters. Forensics Adapter (Cui et al., 2025) inserts adapters into CLIP, and IAPL (Li et al., 2026) adds prompts that are adjusted for each test image. To preserve pretrained knowledge, Effort (Yan et al., 2025b) keeps the main components of the pretrained weights fixed, and DGS-Net (Yan et al., 2026) removes harmful gradient directions. These works mainly improve how forensic features are extracted. Few works study how these features are aggregated into the image-level representation (Section 2.3).

2.2 Semantic Shortcut Suppression

Semantic shortcut occurs when detectors exploit label-related image rather than synthesis artifacts. Existing methods mainly explore three directions: input perturbation, distribution regularization, and subspace projection. Input perturbation methods disrupt the semantic structure of input images. For instance, BSA (Zheng et al., 2024) shuffles image patches. Fu et al. (2025) shuffles and mixes tokens between images, and Zhou et al. (2026) remaps pixel values to break semantic patterns. Distribution regularization methods penalize content correlations across training data. For example, B-Free (Guillaro et al., 2025) builds real and fake images with the same content by inpainting, and DRCT (Chen et al., 2024a) reconstructs real images with a diffusion model. HSIC Bottleneck (Yang et al., 2026a) decouples CLIP features from image captions, and OmniAID (Guo et al., 2026) sends images with different semantic content to different experts. Subspace projection methods separate semantic components in the feature space. For example, GSD (Shuai et al., 2026) finds dominant semantic directions and weakens features along them, and CausalCLIP (Liu et al., 2026) uses learned masks to isolate forgery-irrelevant components. However, existing subspace projections operate at the dataset or image level, overlooking the semantic heterogeneity across different regions within an image. In contrast, our CBA operates at the region level: by clustering patches belonging to the same entity and modeling second-order residual statistics, it thoroughly eliminates local semantic baselines without suffering from global projection bias or linear semantic leakage.

2.3 Forensic Feature Aggregation

Existing methods aggregate forensic evidence at two levels: the patch level, which chooses the image patches to analyze, and the representation level, which combines the token features inside the network. Patch-level methods select image patches with fixed rules. SSP (Chen et al., 2024b) selects the patch with the simplest texture and uses its noise pattern for detection. PatchCraft (Zhong et al., 2024) ranks patches by texture richness and compares the rich-texture and poor-texture patches. TextureCrop (Konstantinidou et al., 2025) crops texture-rich regions from high-resolution images for detection. However, fixed rules may miss the evidence outside the selected patches. Other methods therefore make use of all patches. PPL (Yang et al., 2026b) finds that detectors rely on a few patches, and injects synthetic traces into randomly chosen patches so that the detector learns to use all patches. HiDA-Net (Mu et al., 2026) cuts a high-resolution image into full-resolution local crops, and fuses their features with a downsampled view of the whole image. At the representation level, TAP (Abdullah et al., 2026) learns attention weights to pool all tokens, and Wang et al. (2026) use learnable query tokens to gather forgery evidence from patch tokens instead of the global token. DoU (He et al., 2026) aggregates multi-stage features and prevents complementary cues from collapsing into a few dominant directions. Different from previous methods, our framework works at the cluster level, it groups similar patches and aggregates representation among clusters.

3 Method

3.1 Overall Framework

Given an input image x, we extract features from a frozen CLIP image encoder with layer-wise adapters (Houlsby et al., 2019), and obtain the patch features $\{p_j\}_{j=1}^{N}$, where $p_j \in \mathbb{R}^d$ and N is the number of patches. We then apply K-means (Lloyd, 1982) to the ℓ_2-normalized patch features, which partitions the N patches into K disjoint clusters of semantically similar patches, where $\mathcal{N}_k$ denotes the index set of the k-th cluster.

To suppress the semantics shared within each cluster, CBA subtracts the cluster centroid and extracts a second-order cluster feature $f_k = \mathrm{CBA}\big(\{p_j\}_{j\in\mathcal{N}_k}\big) \in \mathbb{R}^d$ from the centered cluster features. For each class $c \in \mathcal{C} = \{\mathrm{real}, \mathrm{gen}\}$, DES then selects a subset of cluster features that are relevant to c and complementary to each other, and aggregates them into an evidence vector $e_c = \mathrm{DES}\big(\{f_k\}_{k=1}^{K};\, t_c\big) \in \mathbb{R}^d$, where $t_c \in \mathbb{R}^d$ is the randomly initialized, learnable prototype of class c. Finally, the evidence vectors of the two classes are concatenated and fed into the classification head:

$$\hat{y} = \mathrm{Softmax}\big(\psi([e_{\mathrm{real}}; e_{\mathrm{gen}}])\big), \tag{1}$$

where ψ is a linear layer, $[\cdot\,;\cdot]$ denotes concatenation, and $\hat{y}$ denotes the predicted class probabilities.

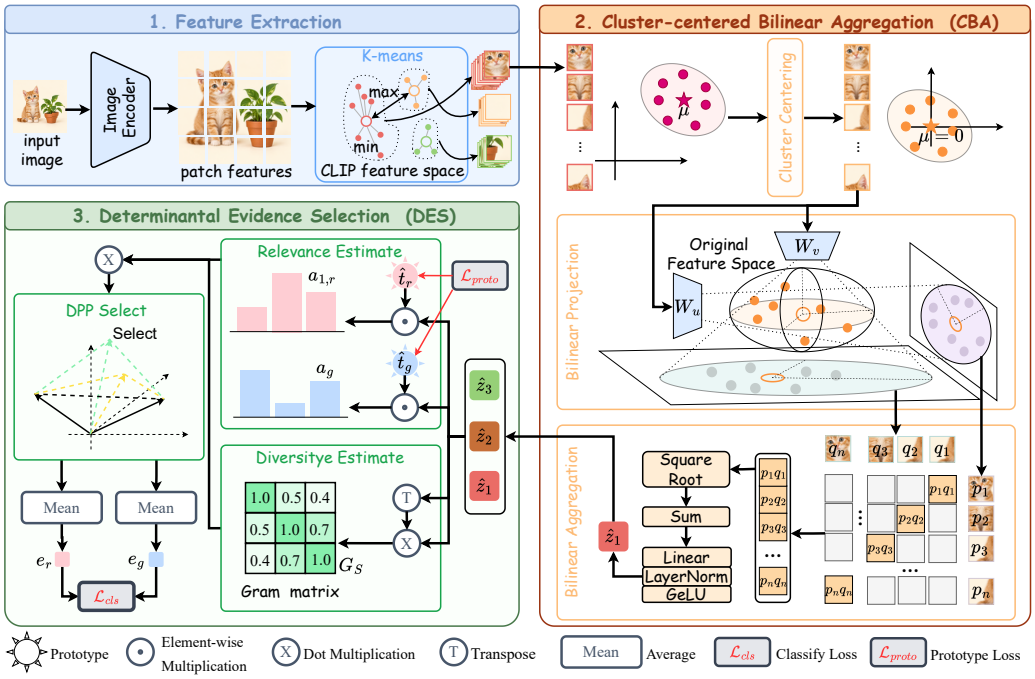

Figure 2: **Overview of AES.** Given an input image x, the model (a) extracts patch features and applies K-means clustering, (b) uses Cluster-centered Bilinear Aggregation (CBA) to center each cluster and model second-order representations, and (c) uses Determinantal Evidence Selection (DES) to select class-relevant and diverse cluster representations for classification.

3.2 Cluster-centered Bilinear Aggregation

Recent methods suppress semantic content at the image level or the dataset level. Patches inside one image nevertheless carry different semantics, so a single image-level operation can distort the pre-trained features. We therefore propose Cluster-centered Bilinear Aggregation (CBA), which groups patches that share similar semantics and extracts second-order statistics, reducing the influence of shared feature components. The module has three stages: cluster centering, bilinear projection, and bilinear aggregation.

Cluster centering. K-means partitions patches by their CLIP features, so patches in one cluster share similar semantics. Based on this observation, we estimate the shared semantic content within each cluster. For each cluster k, we compute the cluster centroid as $\bar{p}_k = \frac{1}{n_k}\sum_{j\in\mathcal{N}_k} p_j$, where $n_k = |\mathcal{N}_k|$. The centroid $\bar{p}_k$ captures the semantic content shared by visually similar patches within cluster k. We therefore subtract $\bar{p}_k$ from each patch feature in the cluster to remove this shared semantic content. The centered residuals are then projected into two distinct subspaces:

$$\tilde{u}_j = W_u(p_j - \bar{p}_k), \qquad \tilde{v}_j = W_v(p_j - \bar{p}_k), \qquad j \in \mathcal{N}_k, \tag{2}$$

where $W_u, W_v \in \mathbb{R}^{r\times d}$ denote the weight parameters of the projection layers.

Bilinear projection. The centered residuals characterize within-cluster feature variations relative to the shared semantic baseline. To read forensic cues from this variation, CBA aggregates these centered representations within the cluster. Computed over all patches without centering, these correlations decompose as $\frac{1}{N}\sum_j p_j p_j^\top = \sum_k \frac{n_k}{N}\left(\Sigma_k + \bar{p}_k\bar{p}_k^\top\right)$, where $\Sigma_k = \frac{1}{n_k}\sum_{j\in\mathcal{N}_k}(p_j-\bar{p}_k)(p_j-\bar{p}_k)^\top$ is the intra-cluster covariance and $\bar{p}_k\bar{p}_k^\top$ is determined by the cluster content. We therefore model Σ_k from the centered residuals, which retains the intra-cluster variation while excluding the content term. Since Σ_k has d^2 entries, we adopt low-rank bilinear pooling (Kim et al., 2017) and compute:

$$g_k = \frac{1}{n_k}\sum_{j\in\mathcal{N}_k} \tilde{u}_j \odot \tilde{v}_j = \mathrm{diag}\left(W_u\Sigma_k W_v^\top\right) \in \mathbb{R}^r, \tag{3}$$

where $\odot$ denotes element-wise multiplication. Each entry of g_k computes the covariance between the corresponding channels of $\tilde{u}_j$ and $\tilde{v}_j$. Therefore, g_k encodes directional cross-channel covariances rather than simple variances, adaptively selecting discriminative forensic correlations.

Bilinear aggregation. The multiplicative interactions in bilinear pooling can produce feature responses with a wide dynamic range. Following bilinear pooling (Lin et al., 2015), we apply signed square-root normalization followed by ℓ_2 normalization to calibrate scales across clusters:

$$h_k = W_o\big(\mathrm{sgn}(g_k) \odot |g_k|^{1/2}\big), \qquad f_k = \mathcal{B}\big(h_k/\|h_k\|_2\big), \tag{4}$$

where the fully-connected layer $W_o \in \mathbb{R}^{d\times r}$ projects the pooled statistics back to dimension d. The block $\mathcal{B}$ consists of Layer Normalization (Ba et al., 2016), a linear layer, and a GELU activation (Hendrycks & Gimpel, 2016). All parameters in CBA are shared across clusters.

In contrast to first-order aggregation that preserves linear semantic pathways, CBA enforces semantic irreversibility through quadratic interactions around the cluster mean. This structure prevents the detector from exploiting reconstructible semantic content and directs attention to subtle generative artifacts. Moreover, the low-rank formulation avoids quadratic matrix computation, ensuring linear computational complexity with respect to patch count. The resulting cluster representations $\{f_k\}_{k=1}^K$ serve as candidate evidence for DES.

3.3 Determinantal Evidence Selection

The cluster representations $\{f_k\}_{k=1}^K$ produced by CBA may differ in task relevance and contain redundant forensic information. Some regions carry few forensic cues and contribute little to the final decision. The detector may also over-rely on a few recurring patterns and then generalize poorly to unseen generators. DES combines class-specific prototype scores with a determinantal diversity criterion (Kulesza & Taskar, 2012) to select informative and complementary descriptors for image-level aggregation. The module comprises three stages: class-guided utility estimation, evidence diversity measurement, and determinantal subset selection with evidence aggregation.

Class-guided task utility estimation. To evaluate whether a cluster representation is diagnostic of generated or not, the feature space requires an explicit frame of reference. We therefore introduce class-specific prototypes $\{t_c\}_{c\in\mathcal{C}}$, where $\mathcal{C} = \{\mathrm{real}, \mathrm{gen}\}$ and $t_c \in \mathbb{R}^d$. These prototypes define separate reference directions , enabling the relevance of each cluster descriptor to be assessed with respect to both classes.

To quantify this relevance, we first project each candidate cluster feature $f_k \in \mathbb{R}^d$ via a linear layer $z_k = W_a f_k + b_a$, with $W_a \in \mathbb{R}^{d\times d}$ and $b_a \in \mathbb{R}^d$. The task utility (relevance score) $a_{k,c} \in [-1, 1]$ of cluster k with respect to class c is measured by the cosine similarity against the corresponding class prototype t_c:

$$a_{k,c} = \hat{z}_k^\top \hat{t}_c, \qquad \hat{z}_k = \frac{z_k}{\|z_k\|_2}, \quad \hat{t}_c = \frac{t_c}{\|t_c\|_2}. \tag{5}$$

Clusters that align closely with the prototype $\hat{t}_c$ achieve higher scores, indicating verified forensic utility for class c and distinguishing them from uninformative background clusters.

Evidence diversity measurement. To prevent the model from selecting mutually redundant clusters that reinforce the same single pattern, we quantify the directional diversity among candidate representations. Let $\hat{f}_k = f_k/\|f_k\|_2$. We construct the Gram matrix $G \in \mathbb{R}^{K\times K}$ where $G_{kk'} = \hat{f}_k^\top \hat{f}_{k'}$. For any candidate subset of clusters $S \subseteq \{1, \ldots, K\}$, let G_S denote the principal submatrix of G indexed by S. The determinant $\det(G_S)$ geometrically measures the squared volume spanned by $\{\hat{f}_k\}_{k\in S}$ in the representation space: it approaches 1 when all features in S are mutually orthogonal, and diminishes toward 0 if collinearity or pattern redundancy exists among them.

Subset selection via DPPs. Balancing task utility against pattern diversity, we formulate the selection energy of a candidate subset S for class c following the quality–diversity decomposition in DPPs (Kulesza & Taskar, 2012):

$$E_c(S) = 2\tau \sum_{k\in S} a_{k,c} + \lambda \log\det\big(G_S + \varepsilon I\big), \tag{6}$$

where $\tau > 0$ is a temperature scaling utility sensitivity, $\lambda > 0$ governs evidence diversity, and $\varepsilon = 10^{-4}$ ensures numerical stability. The first term guarantees that every selected cluster is informative

with respect to the class prototype t_c , while the second term strictly penalizes informational overlap among the chosen features . Formally, when $\lambda = 1$ and $\varepsilon = 0$, Eq. 6 recovers $\log\det(L_S^c)$, the log-determinant of the DPP kernel $L^c = Q_c G Q_c$, where $Q_c = \mathrm{diag}\big(\exp(\tau a_{1,c}), \ldots, \exp(\tau a_{K,c})\big)$.

DES identifies the optimal subset of s clusters ($s < K$) that maximizes the selection energy:

$$S_c^* = \arg\max_{S \in \mathcal{S}_s} E_c(S), \qquad \mathcal{S}_s = \big\{S \subseteq \{1, \ldots, K\} : |S| = s\big\}. \tag{7}$$

Once the optimal subset S_c^* is identified, the representation $e_c \in \mathbb{R}^d$ for class c is obtained by summing selected cluster features:

$$e_c = \frac{1}{s} \sum_{k \in S_c^*} \hat{f}_k. \tag{8}$$

Because the discrete selection in Eq. 7 is non-differentiable, backpropagating gradients through e_c cannot optimize the prototypes t_c or the projection (W_a, b_a). To achieve end-to-end training, we develop a continuous relaxation and an auxiliary prototype loss, as detailed in Section 3.4.

3.4 Training Objective

The complete network is trained end-to-end by minimizing the joint objective:

$$\mathcal{L} = \mathcal{L}_{\mathrm{cls}} + \alpha \mathcal{L}_{\mathrm{proto}}, \tag{9}$$

where α is a balancing hyperparameter (set to 0.1 by default). We supervise the image-level prediction $\hat{y}$ in Eq. 1 using the standard cross-entropy loss:

$$\mathcal{L}_{\mathrm{cls}} = -\log \hat{y}_y. \tag{10}$$

To provide informative gradients for t_c and (W_a, b_a), we replace the hard subset score by its smooth maximum over all candidate subsets $S \in \mathcal{S}_s$:

$$\bar{E}_c = \log\Big(\frac{1}{|\mathcal{S}_s|} \sum_{S \in \mathcal{S}_s} \exp\big(E_c(S)\big)\Big). \tag{11}$$

The prototype loss then requires this smooth maximum of the ground-truth class y to dominate the alternative class:

$$\mathcal{L}_{\mathrm{proto}} = -\log \frac{\exp(\bar{E}_y)}{\sum_{c \in \mathcal{C}} \exp(\bar{E}_c)}. \tag{12}$$

Minimizing $\mathcal{L}_{\mathrm{proto}}$ dynamically rotates the ground-truth prototype t_y toward high-utility, diverse cluster representations while repelling the non-target prototype, with redundant clusters receiving naturally attenuated gradient updates. The complete mathematical derivation and geometric analysis of this gradient dynamic are provided in Appendix A.

4 Experiments

4.1 Experimental Setup

Datasets. We evaluate our method on four benchmarks including GenImage (Zhu et al., 2023), UniversalFakeDetect (UFD) (Ojha et al., 2023), and Synthbuster (Bammey, 2024). Following the protocols of (Zhu et al., 2023; Tan et al., 2025), we train on the SD v1.4 subset of GenImage for GenImage and on four ProGAN classes from ForenSynths (Wang et al., 2020) for UFD, and test on all subsets of each dataset. Synthbuster further covers commercial generators such as DALL·E 3, Firefly, and Midjourney. The in-the-wild dataset, Chameleon (Yan et al., 2025a), comprises images collected from the Internet with high content diversity, thereby better reflecting real-world usage scenarios. Synthbuster and Chameleon are evaluated with the model trained on GenImage. Details of the datasets and compared methods are provided in Appendix B. Appendix C examines whether the learned representation separates authenticity from semantic category.

Evaluation metrics. Following prior studies (Zhu et al., 2023; Ojha et al., 2023; Tan et al., 2025), we adopt accuracy (ACC) as the evaluation metric, using a classification threshold of 0.5, and report

Table 1: The performance on GenImage (Zhu et al., 2023). Best and second-best results are shown in **bold** and underlined, respectively.

Method	Venue	ADM	BigGAN	GLIDE	Midjourney	SD v1.4	SD v1.5	VQDM	Wukong	Avg. ACC
CNNSpot (Wang et al., 2020)	CVPR 2020	50.10	46.80	39.80	52.80	96.30	95.90	53.40	78.60	64.20
UnivFD (Ojha et al., 2023)	CVPR 2023	71.90	90.50	85.40	93.90	96.40	96.20	81.60	94.30	88.80
FreqNet (Tan et al., 2024a)	AAAI 2024	66.80	81.40	86.50	89.60	98.80	98.60	75.80	97.30	86.80
NPR (Tan et al., 2024b)	CVPR 2024	76.90	84.20	89.80	81.00	98.20	97.90	84.10	96.90	88.60
C2P-CLIP (Tan et al., 2025)	AAAI 2025	96.40	**98.70**	**99.00**	88.20	90.90	97.90	96.50	**98.80**	95.80
Effort (Yan et al., 2025b)	ICML 2025	78.70	77.60	93.30	82.40	**99.80**	**99.80**	91.70	97.40	91.10
SAFE (Li et al., 2025)	KDD 2025	82.10	97.80	96.30	95.30	99.40	99.30	96.30	98.20	95.60
PPL (CLIP) (Yang et al., 2026b)	ICLR 2026	94.70	98.00	96.10	94.80	98.50	98.30	**98.50**	98.60	97.20
AES (ours)	—	**96.97**	98.08	98.18	**98.31**	98.11	98.02	98.22	98.35	**98.03**

Table 2: The performance on UFD (Ojha et al., 2023). Best and second-best results are shown in **bold** and underlined, respectively.

		GAN						LDM				GLIDE				
Method	Venue	Pro.	Cyc.	Big.	Sty.	Gau.	Star.	Gui.	L200	L-CFG	L100	G100	G50	G10	DALL-E	Avg. ACC
CNNSpot (Wang et al., 2020)	CVPR 2020	99.99	85.20	70.20	85.70	78.95	91.70	60.07	54.03	54.96	54.14	60.78	63.80	65.66	55.58	70.05
UnivFD (Ojha et al., 2023)	CVPR 2023	**100.0**	98.50	94.50	82.00	**99.50**	97.00	70.03	94.19	73.76	94.36	79.07	79.85	78.14	86.78	87.69
FreqNet (Tan et al., 2024a)	AAAI 2024	97.90	95.84	90.45	**97.55**	90.24	93.41	86.70	84.55	**99.58**	65.56	85.69	**97.40**	88.15	59.06	88.01
FatFormer (Liu et al., 2024)	CVPR 2024	99.9	**99.3**	**99.5**	97.2	99.4	99.8	76.1	98.6	94.9	98.7	94.4	94.7	94.2	98.8	**96.11**
NPR (Tan et al., 2024b)	CVPR 2024	99.84	95.00	87.55	96.23	86.57	99.75	84.55	97.65	98.00	98.20	96.25	97.15	97.35	87.15	94.37
C2P-CLIP (Tan et al., 2025)	AAAI 2025	99.98	97.31	99.12	96.44	99.17	99.60	69.10	**99.25**	97.25	**99.30**	95.25	95.25	96.10	98.55	95.83
PPL (CLIP) (Yang et al., 2026b)	ICLR 2026	89.12	97.16	83.57	89.94	75.29	97.12	**94.55**	98.80	98.80	98.80	**96.90**	**97.40**	**97.65**	98.80	93.85
AES (ours)	—	99.81	95.16	96.85	94.06	97.35	**99.82**	85.25	99.05	95.35	99.20	94.65	94.80	93.55	**98.85**	95.98

the average accuracy over subsets (Avg. ACC). On Chameleon, we additionally report the accuracies on fake and real images. In ablation studies, we also report mAP and mAUC.

Compared methods. We compare our method with twelve detectors: CNNSpot (Wang et al., 2020), UnivFD (Ojha et al., 2023), FatFormer (Liu et al., 2024), FreqNet (Tan et al., 2024a), NPR (Tan et al., 2024b), AIDE (Yan et al., 2025a), C2P-CLIP (Tan et al., 2025), Effort (Yan et al., 2025b), SAFE (Li et al., 2025), AlignedForensics (Rajan et al., 2025), OmniAID (Guo et al., 2026), and PPL (Yang et al., 2026b). Their results are taken from published papers. Details of these methods are summarized in Appendix B.2.

4.2 Implementation Details

We use a frozen CLIP ViT-L/14 (Radford et al., 2021) image encoder and fuse the patch tokens of layers 6, 12, 18, and 24 with layer-wise adapters, as described in Section 3.1. We set the number of clusters to $K = 8$ and the subset size of DES to $s = 2$. We set $\lambda = 0.1$ in Eq. 6 and $\alpha = 0.1$ in Eq. 9. The adapters, CBA, DES, and the classification head are optimized with AdamW (Loshchilov & Hutter, 2019) for one epoch, with a learning rate of 3×10^{-5}, a weight decay of 0.01, and a batch size of 256. No blur, compression, or other data augmentation is applied during training. All experiments are implemented in PyTorch and run on a single NVIDIA RTX 5090 GPU.

4.3 Comparison with State-of-the-Art Methods

We compare our method with representative detectors. GenImage and UFD test cross-generator generalization, while Chameleon and Synthbuster evaluate transfer to in-the-wild and diverse diffusion models, respectively.

On GenImage. As shown in Tab. 1, our method achieves 98.03% Avg. ACC across eight generators. This exceeds the reported SAFE, C2P-CLIP, and PPL values by 2.43, 2.23, and 0.83 percentage points, respectively. Notably, our method maintains balanced performance across diverse generative models, exceeding 96.9% on every generator. ProjectioThese results are consistent with the hypothesis that reducing shared intra-cluster content can limit generator-specific overfitting.

On UFD. As shown in Tab. 2, under the four-class ProGAN protocol, FatFormer attains the highest Avg. ACC of 96.11% across the fourteen generators, and our method follows at 95.98%. FatFormer per-generator accuracies are those reported by Liu et al. (2024); the average is the mean over the fourteen subsets in this table. Our method is more balanced across generators. Accuracy stays above 85% on every subset, at 97.18% on GANs, 94.71% on LDMs, and 95.46% on GLIDE variants. The gap between the strongest and weakest generator is 14.57 percentage points for our method and

Table 3: Cross-dataset accuracy on the in-the-wild dataset Chameleon (Yan et al., 2025a). Best and second-best results are shown in **bold** and underlined, respectively.

	CNNSpot	UnivFD	NPR	AIDE	PPL	OmniAID	**AES**
oACC	60.11	55.62	58.13	62.60	63.94	77.40	**78.72**
Fake/Real	8.86/98.63	17.65/93.50	2.43/**100.0**	20.33/94.38	17.27/99.01	75.60/78.70	**90.25**/70.06

Table 4: Performance on Synthbuster (Bammey, 2024). Best and second-best results are shown in **bold** and underlined, respectively.

Method	Venue	D·E 2	D·E 3	Firefly	GLIDE	MJ	SD 1.3	SD 1.4	SD 2.0	SDXL	Avg. ACC
CNNSpot (Wang et al., 2020)	CVPR 2020	56.50	64.00	54.00	56.40	56.70	97.20	97.00	66.60	66.80	68.40
UnivFD (Ojha et al., 2023)	CVPR 2023	**83.50**	47.40	**89.90**	53.30	52.50	70.40	69.90	75.70	68.00	67.84
NPR (Tan et al., 2024b)	CVPR 2024	51.10	49.30	46.50	48.50	52.80	51.40	51.80	46.00	52.80	50.02
AIDE (Yan et al., 2025a)	ICLR 2025	34.90	33.70	24.80	65.00	57.50	74.10	73.70	53.20	68.40	53.92
AlignedForensics (Rajan et al., 2025)	ICLR 2025	50.20	48.90	51.70	53.50	**98.70**	**98.80**	**98.80**	**98.60**	**97.30**	77.39
C2P-CLIP (Tan et al., 2025)	AAAI 2025	55.60	63.20	59.50	**86.70**	52.90	75.20	76.70	69.20	77.70	68.52
Effort (Yan et al., 2025b)	ICML 2025	67.90	79.30	56.10	79.00	77.20	81.60	81.70	77.90	80.50	75.69
SAFE (Li et al., 2025)	KDD 2025	58.00	9.90	10.30	52.20	56.70	59.40	59.10	53.00	59.50	46.46
AES (ours)	—	81.35	**89.65**	82.90	76.15	84.45	89.25	89.40	82.05	85.55	**84.53**

23.7 percentage points for FatFormer. On Guided Diffusion, the weakest subset for both methods, our method reaches 85.25%, while FatFormer, UnivFD, and C2P-CLIP reach 76.1%, 70.03%, and 69.10%, respectively.

On Chameleon. As shown in Tab. 3, on real-world unconstrained imagery, our method achieves 78.72% oACC, exceeding OmniAID and PPL by 1.32 and 14.78 percentage points, respectively. Several prior detectors report low fake-image accuracy: 2.43% for NPR, 17.65% for UnivFD, and 20.33% for AIDE. Our method achieves 90.25% fake-image accuracy and 70.06% real-image accuracy.

On Synthbuster. As shown in Tab. 4, our method establishes competitive performance across nine diffusion models with 84.53% Avg. ACC, outperforming AlignedForensics (77.39%) and Effort (75.69%). On DALL·E 2 and Firefly, AlignedForensics reports 50.20% and 51.70% accuracy, whereas our method reaches 81.35% and 82.90%. These results suggest that the cluster-centered representation can transfer across the tested diffusion models.

4.4 Ablation Studies

We next examine the contribution of the proposed modules. Unless otherwise stated, each variant uses the same frozen backbone, training protocol, and evaluation split.

Effects of Key Modules. We evaluate the five meaningful configurations. Configurations with semantic clustering disabled but CBA or DES enabled are omitted because both modules operate on cluster features. As shown in Tab. 5, adding semantic clustering improves Avg. ACC from 93.88% to 94.62%. Adding DES produces a larger gain, reaching 97.10%, while adding CBA reaches 96.68%. Combining both modules gives the best result, with 98.03% Avg. ACC, 99.81% mAP, and 99.80% mAUC.

Key Components in CBA. We compare L_1-norm pooling with max pooling in CBA. L_1-norm pooling reaches 83.45% ACC, 95.92% AP, and 95.94% AUC, while max pooling obtains 83.37%, 94.17%, and 94.18%, respectively. L_1-norm pooling improves AP and AUC by 1.75 and 1.76 percentage points, respectively.

Key Components in DES. We isolate the relevance and diversity terms in DES. Relevance-only selection reaches 97.16% Avg. ACC, 99.73% mAP, and 99.71% mAUC, and adding the diversity term further improves Avg. ACC to 98.03%. Diversity-only selection reaches 94.85% Avg. ACC, whereas non-selective baselines that pool all cluster descriptors without selection (average and max pooling) obtain 93.97% and 92.72%, respectively. These results identify class relevance as the primary signal for selecting useful evidence, with diversity acting as a complementary constraint.

Sensitivity of K and s. Table 8 shows that the selected setting $K = 8$, $s = 2$ gives the best result. At $s = 2$, the tested settings with fewer or more than eight clusters yield lower Avg. ACC.

Table 5: Component-wise ablation study.

Models	Avg. ACC (%)	mAP (%)	mAUC (%)
Baseline	93.88	99.23	99.24
Baseline + Semantic Clustering	94.62	99.39	99.41
Baseline + Clustering + DES	97.10	99.56	99.55
Baseline + Clustering + CBA	96.68	99.45	99.57
Full AES	**98.03**	**99.81**	**99.80**

Table 6: Ablation of different pooling operations in CBA.

Pooling variant	ACC (%)	AP (%)	AUC (%)
L_1-norm pooling	83.45	95.92	95.94
Max pooling	83.37	94.17	94.18

Increasing the selected subset beyond two candidates also reduces accuracy, which supports using a compact subset of complementary evidence. We therefore use $K = 8$, $s = 2$, and $\lambda = 0.1$.

Table 7: Ablation of different selection objectives in DES.

Selection variant	Avg. ACC (%)	mAP (%)	mAUC (%)
Relevance-only selection	97.16	99.73	99.71
Diversity-only selection	94.85	99.62	99.60
All clusters (avg pooling)	93.97	99.41	99.34
All clusters (max pooling)	92.72	99.59	99.56
Relevance + diversity	**98.03**	**99.81**	**99.80**

Table 8: Sensitivity of semantic clusters K and the selected subset size s.

K	$s = 1$	$s = 2$	$s = 3$	$s = 4$
4	94.41	94.66	93.30	92.34
6	96.91	97.41	95.95	94.75
8	97.01	**98.03**	96.49	95.24
10	96.87	97.55	96.47	95.21

5 Conclusion

We presented AES, a readout framework for generalizable AI-generated image detection that replaces the content-dominated [CLS] summary with residual and diverse forensic evidence. CBA groups semantically similar patch tokens, removes their shared cluster content, and models the remaining second-order interactions. DES then selects a compact subset of cluster descriptors by jointly modeling class relevance and feature-direction diversity. Across four benchmarks, AES achieves 98.03% Avg. ACC on GenImage, 95.98% Avg. ACC on UniversalFakeDetect, 78.72% oACC on Chameleon, and 84.53% Avg. ACC on Synthbuster. Component ablations show that semantic clustering, CBA, and DES provide complementary gains, while the sensitivity study selects eight clusters and two evidence descriptors as the default configuration. These results support residual feature extraction and class-conditioned evidence selection as an effective way to improve cross-generator generalization.

AI USE STATEMENT

In this work, we used generative AI tools to assist with translation and implement methods, formulate mathematical claims. We have not used generative AI tools to design or provide feedback on research methodology or experiments, propose or refine hypotheses, support qualitative and thematic data analysis, or interpret results. Generating synthetic datasets, helping develop theoretical models or conceptual frameworks, providing critical ingredients for proving mathematical claims, assisting in the writing of proofs, and cleaning and reformatting datasets are not applicable to this work. Additionally, we used generative AI tools to create or edit software code and to draft and edit the research paper, including writing the text and handling the TeX formatting. We have reviewed all AI-assisted work. LLM-generated code and text were checked by at least two people. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

The main methodological and implementation details of AES are reported in the paper. The algorithmic specification is given in Section 3. The evaluation protocol and implementation details are given in Sections 4.1 and 4.2, respectively. All experiments use fixed random seeds. Secondary implementation details, the full experimental environment, training scripts, and pretrained weights will be released to facilitate reproduction of the reported results.

REFERENCES

Ahmed Abdullah, Nikolas Ebert, and Oliver Wasenmüller. TAP into the patch tokens: Leveraging vision foundation model features for AI-generated image detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pp. 379–389, 2026. SAFE Workshop.

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

Quentin Bammey. Synthbuster: Towards detection of diffusion model generated images. *IEEE Open Journal of Signal Processing*, 5:1–9, 2024. doi: 10.1109/OJSP.2023.3337714.

Baoying Chen, Jishen Zeng, Jianquan Yang, and Rui Yang. DRCT: Diffusion reconstruction contrastive training towards universal detection of diffusion generated images. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, volume 235 of *Proceedings of Machine Learning Research*, pp. 7621–7639, 2024a.

Jiaxuan Chen, Jieteng Yao, and Li Niu. A single simple patch is all you need for AI-generated image detection, 2024b.

Xinjie Cui, Yuezun Li, Ao Luo, Jiaran Zhou, and Junyu Dong. Forensics adapter: Adapting CLIP for generalizable face forgery detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19207–19217, 2025.

Jia Deng, Wei Dong, Richard Socher, Kai-Li Li, and Fei-Fei Li. ImageNet: A Large-Scale Hierarchical Image Database. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 248–255, 2009.

Xinghe Fu, Zhiyuan Yan, Taiping Yao, Shen Chen, and Xi Li. Exploring Unbiased Deepfake Detection via Token-Level Shuffling and Mixing. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2025. arXiv:2501.04376.

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

Yuncheng Guo, Junyan Ye, Chenjue Zhang, Hengrui Kang, Haohuan Fu, Conghui He, and Weijia Li. OmniAID: Decoupling Semantics and Artifacts for Universal AI-Generated Image Detection in the Wild. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2026. arXiv:2511.08423.

Qinghui He, Haifeng Zhang, Qiao Qin, Bo Liu, Xiuli Bi, and Bin Xiao. Diversity over uniformity: Rethinking representation in generated image detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 40407–40417, 2026.

Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (GELUs). *arXiv preprint arXiv:1606.08415*, 2016.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for NLP. In *International Conference on Machine Learning (ICML)*, pp. 2790–2799, 2019.

Jin-Hwa Kim, Kyoung-Woon On, Woosang Lim, Jeonghee Kim, Jung-Woo Ha, and Byoung-Tak Zhang. Hadamard product for low-rank bilinear pooling. In *International Conference on Learning Representations (ICLR)*, 2017.

Despina Konstantinidou, Christos Koutlis, and Symeon Papadopoulos. TextureCrop: Enhancing synthetic image detection through texture-based cropping. In *Proceedings of the Winter Conference on Applications of Computer Vision (WACV) Workshops*, pp. 1459–1468, 2025.

Christos Koutlis and Symeon Papadopoulos. Leveraging Representations from Intermediate Encoder-blocks for Synthetic Image Detection. In *Computer Vision – ECCV (European Conference on Computer Vision)*, 2024. arXiv:2402.19091.

Alex Kulesza and Ben Taskar. Determinantal point processes for machine learning. *Foundations and Trends in Machine Learning*, 5(2–3):123–286, 2012. doi: 10.1561/2200000044.

Ouxiang Li, Jiayin Cai, Yanbin Hao, Xiaolong Jiang, Yao Hu, and Fuli Feng. Improving synthetic image detection towards generalization: An image transformation perspective. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, pp. 2405–2414, 2025. doi: 10.1145/3690624.3709392.

Yiheng Li, Zichang Tan, Guoqing Xu, Zhen Lei, Xu Zhou, and Yang Yang. Towards Generalizable AI-Generated Image Detection via Image-Adaptive Prompt Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026. arXiv:2508.01603.

Tsung-Yu Lin, Aruni RoyChowdhury, and Subhransu Maji. Bilinear CNN models for fine-grained visual recognition. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 1449–1457, 2015.

Bo Liu, Qiao Qin, and Qinghui He. CausalCLIP: Causally-informed feature disentanglement and filtering for generalizable detection of generated images. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, volume 40, pp. 7069–7077, 2026. doi: 10.1609/aaai.v40i9.37642.

Huan Liu, Zichang Tan, Chuangchuang Tan, Yunchao Wei, Jingdong Wang, and Yao Zhao. Forgery-aware adaptive transformer for generalizable synthetic image detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10770–10780, 2024.

Stuart P. Lloyd. Least squares quantization in PCM. *IEEE Transactions on Information Theory*, 28 (2):129–137, 1982.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations (ICLR)*, 2019.

Lianrui Mu, Haoji Hu, Xingze Zou, Jianhong Bai, and Jiaqi Hu. No pixel left behind: A detail-preserving architecture for robust high-resolution AI generated image detection. In *International Conference on Learning Representations (ICLR)*, 2026.

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

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *Proceedings of the International Conference on Machine Learning (ICML)*, pp. 8748–8763, 2021.

Ani Sundar Rajan, Utkarsh Ojha, Jonas Schloesser, and Yong Jae Lee. Aligned Datasets Improve Detection of Latent Diffusion-Generated Images. In *International Conference on Learning Representations (ICLR)*, 2025.

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

Chuangchuang Tan, Yao Zhao, Shikui Wei, Guanghua Gu, Ping Liu, and Yunchao Wei. Frequency-aware deepfake detection: Improving generalizability through frequency space domain learning. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, volume 38, pp. 5052–5060, 2024a. doi: 10.1609/aaai.v38i5.28310.

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 (CVPR)*, pp. 28130–28139, 2024b.

Chuangchuang Tan, Renshuai Tao, Huan Liu, Guanghua Gu, Baoyuan Wu, Yao Zhao, and Yunchao Wei. C2P-CLIP: Injecting category common prompt in CLIP to enhance generalization in deepfake detection. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, volume 39, pp. 7184–7192, 2025. doi: 10.1609/aaai.v39i7.32772.

Changshuo Wang, Jiangming Wang, Ke-Yue Zhang, Taiping Yao, Shouhong Ding, Shunli Wang, Ran Yi, and Lizhuang Ma. Beyond [CLS] token: Query-driven token-level forgery purification for generalizable deepfake detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 42922–42931, 2026.

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 (CVPR)*, pp. 8695–8704, 2020.

Jiazhen Yan, Ziqiang Li, Fan Wang, Boyu Wang, Ziwen He, and Zhangjie Fu. DGS-Net: Distillation-Guided Gradient Surgery for CLIP Fine-Tuning in AI-Generated Image Detection. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2026. arXiv:2511.13108.

Shilin Yan, Ouxiang Li, Jiayin Cai, Yanbin Hao, Xiaolong Jiang, Yao Hu, and Weidi Xie. A sanity check for AI-generated image detection. In *International Conference on Learning Representations (ICLR)*, 2025a.

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. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, volume 267 of *Proceedings of Machine Learning Research*, pp. 70268–70288, 2025b.

Chin-Chia Yang, Yung-Yu Chuang, Hwann-Tzong Chen, and Tyng-Luh Liu. HSIC bottleneck for cross-generator and domain-incremental synthetic image detection. In *International Conference on Learning Representations (ICLR)*, 2026a.

Zheng Yang, Ruoxin Chen, Zhiyuan Yan, Keyue Zhang, Xinghe Fu, Shuang Wu, Xiujun Shu, Taiping Yao, Shouhong Ding, Zequn Qin, and Xi Li. All patches matter, more patches better: Enhance AI-generated image detection via panoptic patch learning. In *International Conference on Learning Representations (ICLR)*, 2026b.

Chende Zheng, Chenhao Lin, Zhengyu Zhao, Hang Wang, Xu Guo, Shuai Liu, and Chao Shen. Breaking semantic artifacts for generalized AI-generated image detection. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 59570–59596, 2024.

Nan Zhong, Yiran Xu, Sheng Li, Zhenxing Qian, and Xinpeng Zhang. PatchCraft: Exploring texture patch for efficient AI-generated image detection, 2024.

Chenming Zhou, Jiaan Wang, Yu Li, Lei Li, Juan Cao, and Sheng Tang. Beyond Semantic Features: Pixel-level Mapping for Generalized AI-Generated Image Detection. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2026. arXiv:2512.17350.

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. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 77771–77782, 2023. Datasets and Benchmarks Track.

A Mathematical Derivation of Prototype Optimization

In this section, we provide the complete mathematical derivation of the continuous relaxation and the analytical gradient for the prototype loss $\mathcal{L}_{\text{proto}}$ introduced in Section 3.4.

A.1 Continuous Relaxation and Marginal Selection Probability

Recall that for each candidate subset $S \in \mathcal{S}_s$, its selection score for class $c \in \mathcal{C}$ is given by Eq. 6:

$$E_c(S) = 2\tau \sum_{k \in S} a_{k,c} + \lambda \log\det\big(G_S + \varepsilon I\big), \tag{13}$$

where $a_{k,c} = \hat{z}_k^\top \hat{t}_c$, $\hat{z}_k = z_k / \|z_k\|_2$, $z_k = W_a f_k + b_a$, and $\hat{t}_c = t_c / \|t_c\|_2$. The Gibbs distribution $\pi_c(S)$ over candidate subsets $\mathcal{S}_s$ is defined as:

$$\pi_c(S) = \frac{\exp\big(E_c(S)\big)}{\sum_{S' \in \mathcal{S}_s} \exp\big(E_c(S')\big)}. \tag{14}$$

The marginal probability that a specific cluster $k \in \{1, \dots, K\}$ is included in the selected subset of size s under class c is given by:

$$\rho_{k,c} = \sum_{S \in \mathcal{S}_s,\, k \in S} \pi_c(S). \tag{15}$$

A.2 Analytical Gradient of the Prototype Loss

The prototype loss is defined via the smooth maximum $\bar{E}_c = \log\big(\frac{1}{|\mathcal{S}_s|} \sum_{S \in \mathcal{S}_s} \exp(E_c(S))\big)$:

$$\mathcal{L}_{\text{proto}} = -\log \frac{\exp(\bar{E}_y)}{\sum_{c' \in \mathcal{C}} \exp(\bar{E}_{c'})} = -\bar{E}_y + \log \sum_{c' \in \mathcal{C}} \exp(\bar{E}_{c'}), \tag{16}$$

where $y \in \mathcal{C}$ denotes the ground-truth label. Let σ_c denote the softmax probability over the smooth maximum scores:

$$\sigma_c = \frac{\exp(\bar{E}_c)}{\sum_{c' \in \mathcal{C}} \exp(\bar{E}_{c'})}. \tag{17}$$

Differentiating $\mathcal{L}_{\text{proto}}$ with respect to the smooth maximum $\bar{E}_c$ yields:

$$\frac{\partial \mathcal{L}_{\text{proto}}}{\partial \bar{E}_c} = -\delta_{cy} + \sigma_c = -(\delta_{cy} - \sigma_c), \tag{18}$$

where δ_{cy} is the Kronecker delta ($\delta_{cy} = 1$ if $c = y$, and 0 otherwise). Next, differentiating $\bar{E}_c$ with respect to the subset score $E_c(S)$ gives:

$$\frac{\partial \bar{E}_c}{\partial E_c(S)} = \frac{\exp(E_c(S))}{\sum_{S' \in \mathcal{S}_s} \exp(E_c(S'))} = \pi_c(S). \tag{19}$$

Since only the relevance term in $E_c(S)$ depends on $a_{k,c}$, we have:

$$\frac{\partial E_c(S)}{\partial a_{k,c}} = 2\tau \cdot \mathbb{I}(k \in S), \tag{20}$$

where $\mathbb{I}(\cdot)$ is the indicator function. Applying the chain rule along with Eq. 19 and Eq. 15:

$$\frac{\partial \bar{E}_c}{\partial a_{k,c}} = \sum_{S \in \mathcal{S}_s} \frac{\partial \bar{E}_c}{\partial E_c(S)} \frac{\partial E_c(S)}{\partial a_{k,c}} = 2\tau \sum_{S \in \mathcal{S}_s,\, k \in S} \pi_c(S) = 2\tau \rho_{k,c}. \tag{21}$$

Furthermore, the derivative of the cosine similarity $a_{k,c} = \hat{z}_k^\top \hat{t}_c$ with respect to the unnormalized prototype vector t_c is:

$$\frac{\partial a_{k,c}}{\partial t_c} = \frac{1}{\|t_c\|_2}\Big(I - \hat{t}_c \hat{t}_c^\top\Big)\hat{z}_k = \frac{1}{\|t_c\|_2}\big(\hat{z}_k - a_{k,c}\hat{t}_c\big). \tag{22}$$

Finally, combining Eq. 18, $\frac{\partial \bar{E}_c}{\partial a_{k,c}}$, and Eq. 22 via the chain rule, the analytical gradient evaluates to:

$$\frac{\partial \mathcal{L}_{\text{proto}}}{\partial t_c} = \frac{\partial \mathcal{L}_{\text{proto}}}{\partial \bar{E}_c} \sum_{k=1}^{K} \frac{\partial \bar{E}_c}{\partial a_{k,c}} \frac{\partial a_{k,c}}{\partial t_c} = -\frac{2\tau\,(\delta_{cy} - \sigma_c)}{\|t_c\|_2} \sum_{k=1}^{K} \rho_{k,c}\big(\hat{z}_k - a_{k,c}\hat{t}_c\big). \tag{23}$$

Reversing the sign yields the exact gradient descent direction:

$$-\frac{\partial \mathcal{L}_{\text{proto}}}{\partial t_c} = \frac{2\tau\,(\delta_{cy} - \sigma_c)}{\|t_c\|_2} \sum_{k=1}^{K} \rho_{k,c}\big(\hat{z}_k - a_{k,c}\hat{t}_c\big). \tag{24}$$

A.3 Geometric Analysis of Optimization Dynamics

Eq. 24 reveals three desirable geometric properties that govern prototype learning:

1. **Orthogonal Directional Updates**: Since $\hat{t}_c^\top(\hat{z}_k - a_{k,c}\hat{t}_c) = a_{k,c} - a_{k,c} = 0$, the update vector $\hat{z}_k - a_{k,c}\hat{t}_c$ is strictly orthogonal to the current prototype direction $\hat{t}_c$. Thus, gradient descent modifies only the angular orientation of the prototype on the unit hypersphere without altering feature magnitude. For the ground-truth class ($c = y$), $\delta_{cy} - \sigma_c > 0$, which rotates t_y toward the relevant cluster representations; for the non-target class ($c \neq y$), $\delta_{cy} - \sigma_c = -\sigma_c < 0$, actively repelling t_c away from them.
2. **Diversity-attenuated Attribution**: Each cluster feature $\hat{z}_k$ contributes in exact proportion to its marginal selection probability $\rho_{k,c}$. Under the determinantal formulation, subsets containing collinear features yield near-zero determinants $\det(G_S)$, severely penalizing their selection probabilities $\pi_c(S)$. Consequently, redundant features receive negligible marginal attribution $\rho_{k,c}$, ensuring that mutually orthogonal and diverse representations dominate the gradient update.
3. **Adaptive Gradient Attenuation Modulated by Prediction Confidence**: The update magnitude for both classes is bounded by $|\delta_{cy} - \sigma_c| \leq 1 - \sigma_y$. As the network becomes confident in distinguishing the correct forensic evidence ($\sigma_y \to 1$), the gradient smoothly attenuates to zero. This dynamic naturally prevents gradient instability and protects the prototypes from overfitting to individual training samples.

B Experimental Setup Details

B.1 Details of Datasets

We describe the datasets used for training and evaluation in our experiments. For the number of images and the generation settings of each subset, we refer readers to the original papers.

Training data. We use two training sets.

- **GenImage SD v1.4.** The SD v1.4 subset of GenImage (Zhu et al., 2023) contains real images from ImageNet (Deng et al., 2009) and images generated by SD v1.4. We use it for the experiments on GenImage, Synthbuster, and Chameleon.

- **ProGAN.** Following (Tan et al., 2025), we use four classes (car, cat, chair, and horse) of the ProGAN training set in ForenSynths (Wang et al., 2020), whose real images are from LSUN. We use it for the experiments on UFD.

Evaluation data. We evaluate on three benchmarks in which each subset comes from a known generator, and on one in-the-wild dataset.

- **GenImage** (Zhu et al., 2023) contains eight subsets generated by ADM, BigGAN, GLIDE, Midjourney, SD v1.4, SD v1.5, VQDM, and Wukong. The real images of all subsets are from ImageNet.
- **UniversalFakeDetect** (Ojha et al., 2023) contains fourteen subsets. The six GAN subsets, ProGAN, CycleGAN, BigGAN, StyleGAN, GauGAN, and StarGAN, are from ForenSynths (Wang et al., 2020). The other eight subsets are collected by Ojha et al. (2023), including Guided Diffusion, LDM with 200 steps, 200 steps with classifier-free guidance, and 100 steps, GLIDE with three step settings (100-27, 50-27, and 100-10), and DALL·E.
- **Synthbuster** (Bammey, 2024) contains 1,000 images for each of nine generators, namely DALL·E 2, DALL·E 3, Adobe Firefly, GLIDE, Midjourney, SD 1.3, SD 1.4, SD 2, and SD XL, together with 1,000 real images from RAISE.
- **Chameleon** (Yan et al., 2025a) contains AI-generated images collected from online AI art communities, which are selected to be difficult for human perception, and real images collected from the Internet.

These datasets cover GANs, diffusion models, and commercial generators, as well as images shared on the Internet.

B.2 Compared Methods

We summarize each compared method in one sentence below, grouped by backbone.

CNN-based detectors.

- **CNNSpot** (Wang et al., 2020) trains a ResNet-50 classifier with blur and JPEG augmentation.
- **FreqNet** (Tan et al., 2024a) extracts high-frequency components with an FFT-based high-pass filter and learns in the frequency domain with learnable amplitude and phase adjustments.
- **NPR** (Tan et al., 2024b) computes neighboring pixel relationships to capture the local artifacts introduced by up-sampling layers.
- **SAFE** (Li et al., 2025) is a lightweight detector that replaces resizing with cropping, adds ColorJitter and RandomRotation augmentation, and applies patch-based random masking.
- **AlignedForensics** (Rajan et al., 2025) trains a ResNet-50 on real images and their reconstructions by the autoencoder of a latent diffusion model, so that each real and fake training pair differs only in decoder artifacts.

VFM-based detectors.

- **UnivFD** (Ojha et al., 2023) trains a linear classifier on frozen CLIP ViT-L/14 features.
- **FatFormer** (Liu et al., 2024) adapts CLIP with a forgery-aware adapter that fuses image and frequency features and aligns them with text prompts.
- **AIDE** (Yan et al., 2025a) combines a semantic branch based on CLIP ConvNeXt with a patch-wise branch that selects patches by their spectral energy.
- **C2P-CLIP** (Tan et al., 2025) injects category common prompts into the CLIP image encoder through contrastive learning.
- **Effort** (Yan et al., 2025b) decomposes the pretrained weights with SVD, keeps the principal components fixed, and learns forgery cues in the remaining subspace.

- **OmniAID** (Guo et al., 2026) builds a mixture of experts on CLIP ViT-L/14 and routes images with different semantic content to different experts to decouple semantics from artifacts.
- **PPL** (Yang et al., 2026b) fine-tunes CLIP with LoRA using randomized patch reconstruction and patch-wise contrastive learning, so that the detector uses artifacts from all patches.
- **IAPL** (Li et al., 2026) inserts learnable tokens into the CLIP image encoder and adapts them to each test image through test-time token tuning.

IAPL and PGC are not in the benchmark tables. They appear in the representation analysis below. PGC is reported under that name in Fig. 4.

C Visualization and Representation Analysis

This section asks whether detector representations separate authenticity from object category. Every detector is trained on GenImage SD v1.4. Real images come from ImageNet (Deng et al., 2009), and synthetic images come from ProGAN (Wang et al., 2020).

Fig. 3 projects these representations for all twenty ProGAN categories. Frozen CLIP, C2P-CLIP, and Effort cluster by category in every group, and real and generated markers of the same category occupy the same cluster. IAPL keeps much of this category structure. AIDE mixes categories more than these models, but real and generated markers still overlap. Our representation splits every group into a real population and a generated population, with the five category colors intermixed inside each population.

Fig. 4 measures the same distinction with a linear probe on all twenty ProGAN classes, using 100 real and 100 generated test images per class. The probe is trained for 50 epochs with mean squared error to predict the raw CLIP ViT-L/14 image embedding from the detector feature immediately before the binary head. The predicted embedding is matched by cosine similarity to 20 fixed CLIP text prototypes. No 20-way classifier is trained. Direct classification with pretrained CLIP features reaches 85.82%. At epoch 0, every randomly initialized probe is near the 5% chance level. After 50 epochs, C2P-CLIP, Effort, PGC, and IAPL recover 30.6%, 41.1%, 49.0%, and 14.2% accuracy, respectively, whereas our representation remains at 5.0%. The probe therefore reads category semantics from these baseline features and cannot read them from our residual representation. This agrees with Fig. 3: those features stay organized by category, whereas ours is organized by authenticity.

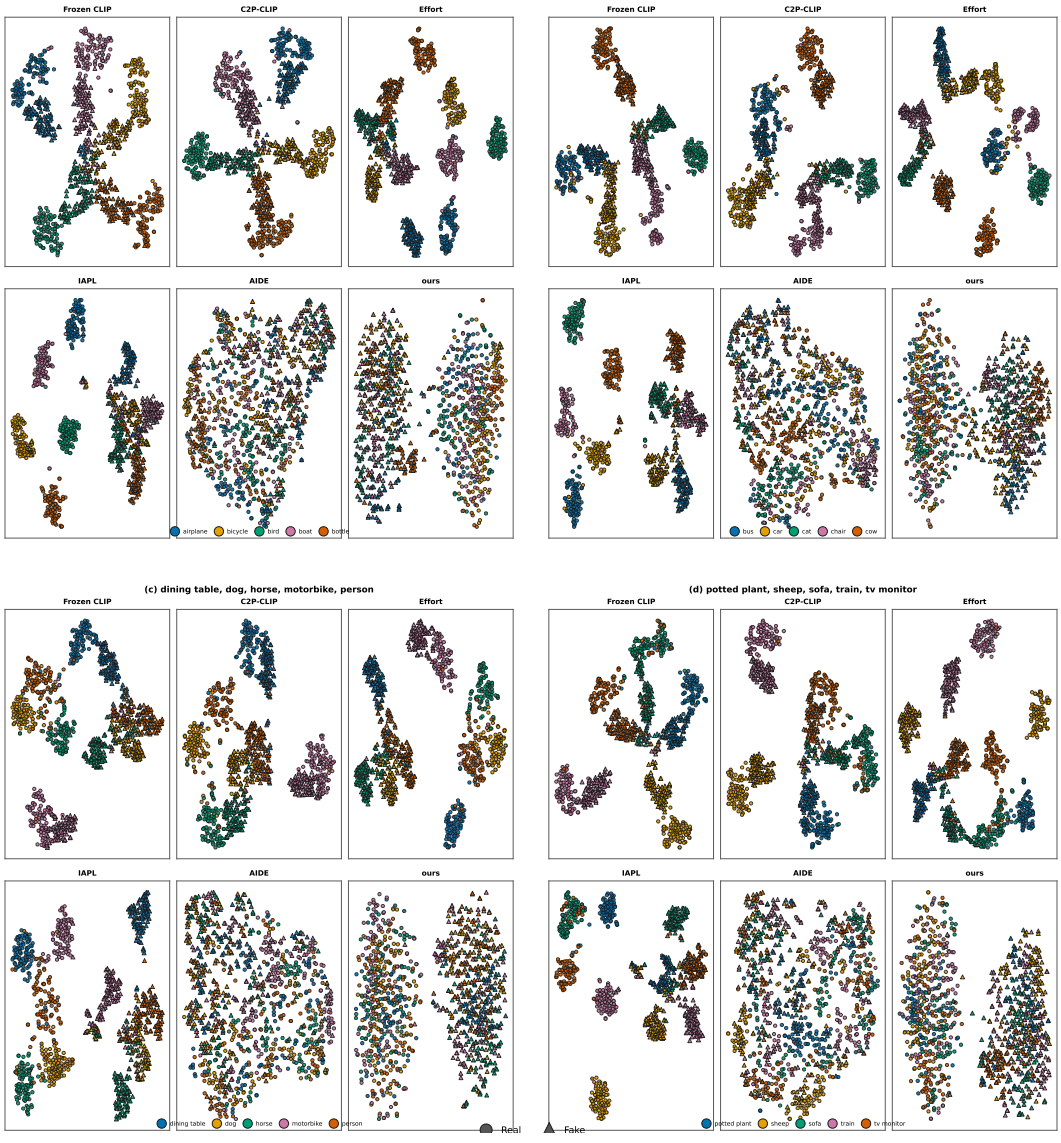

Figure 3: t-SNE on all twenty ProGAN categories, in four groups of five. Circles denote real images and triangles denote ProGAN images. Colors denote semantic categories. Each panel is fitted independently.

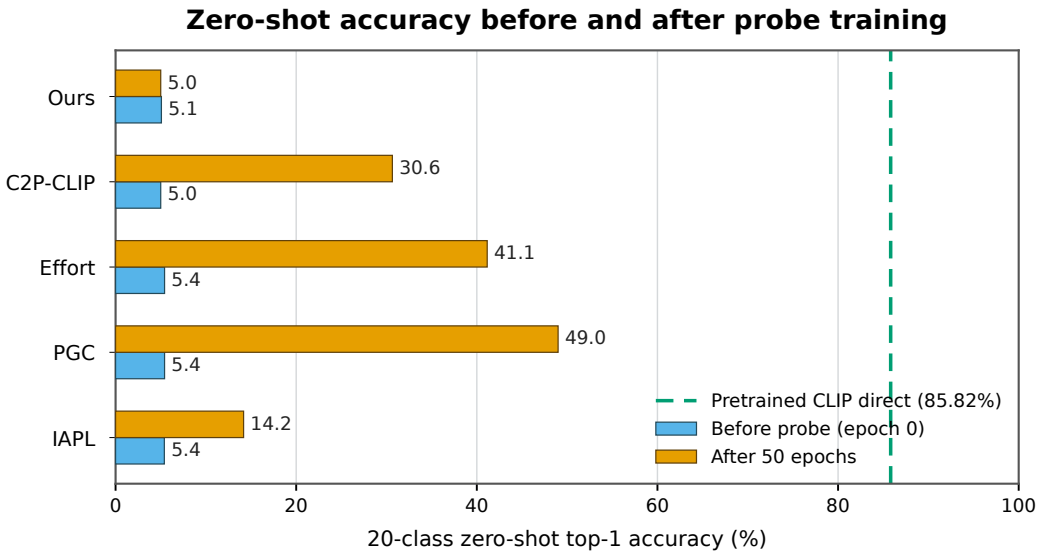

Figure 4: 20-class zero-shot top-1 accuracy before and after fitting a linear probe that maps each detector representation to the frozen CLIP ViT-L/14 image embedding. The dashed line is direct zero-shot classification with pretrained CLIP features (85.82%). Chance accuracy on 20 classes is 5%. AIDE is not included in this plot.