DETECTOR-GUIDED FAKE SYNTHESIS ENHANCEMENT FOR GENERALIZABLE AI-GENERATED IMAGE DETECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

AI-generated image (AIGI) detectors often generalize poorly to unseen generators and image domains because they may exploit shortcut cues specific to the training distribution. Recent data-alignment methods reduce real-fake discrepancies but can introduce new pattern-specific artifacts through pixel-space synthesis. To address this limitation, we propose a feature-space fake synthesis framework that constructs additional fake-labeled training samples by interpolating aligned real and synthetic representations, avoiding additional pixel-level transformations while producing challenging fake features closer to the real distribution. We further introduce detector-guided adversarial optimization, where detector learning and fake-feature synthesis are alternated so that the evolving detector guides the generation of informative hard examples. To enable efficient adversarial optimization and pair-adaptive fake synthesis, we design a lightweight mixing coefficient generator that predicts a bounded interpolation coefficient for each real-fake pair. Additionally, we introduce a coefficient diversity constraint to mitigate generator collapse during adversarial training. Extensive experiments on nine benchmarks demonstrate improved generalization to unseen generators and image domains.

1 INTRODUCTION

Rapid advances in generative models have enabled the widespread creation of highly realistic synthetic images, raising concerns about visual misinformation, fraud, and digital-content authenticity. Rombach et al. (2022) These concerns have motivated research on AI-generated image (AIGI) detection, which aims to distinguish synthetic images from real ones Wang et al. (2020). Although detectors are typically trained on images from a limited set of generators and domains, deployment requires them to handle images from unseen generators and domains. Existing detectors often exhibit substantial performance degradation under these distribution shifts, making generalization beyond the training distribution a central challenge in AIGI detection Ojha et al. (2023); Wang et al. (2023); Tan et al. (2024).

Existing methods seek transferable forensic cues in spatial artifacts, frequency statistics, reconstruction behavior, or pretrained visual representations Frank et al. (2020); Wang et al. (2023); Tan et al. (2024); Ojha et al. (2023); Cozzolino et al. (2024). However, real and synthetic images in the training data may also exhibit systematic differences in semantic content, resolution, file format, or compression statistics, creating label correlations that do not necessarily reflect the generation process. Such differences may be more readily learned than subtle generation-related traces, leading detectors to rely on shortcuts that limit generalization beyond the training distribution Zheng et al. (2024); Rajan et al. (2025).

Recent data-alignment methods address this problem by constructing synthetic counterparts of real images to obtain more closely aligned real–fake training pairs. AlignedForensics Rajan et al. (2025) reconstructs real images using the autoencoder of a latent diffusion model, largely preserving their semantic content and input-level attributes. DRCT Chen et al. (2024) uses diffusion reconstruction to generate hard synthetic samples that closely resemble real images and incorporates contrastive learning to promote sensitivity to subtle generation artifacts. DDA Chen et al. (2025) further reduces discrepancies through joint alignment in the pixel and frequency domains. Although these methods

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.

partially mitigate misalignment between real and fake training images, constructing pixel-level fake images still introduce pattern-specific artifacts, such as characteristic VAE reconstruction patterns and pixel-level ghosting. Detectors may exploit these artifacts as shortcuts, limiting their ability to generalize to unseen generators.

To mitigate the adverse effects of pattern-specific artifacts in pixel-level fake images, we construct additional fake-labeled training samples by mixing real and synthetic image representations directly in feature space. This approach augments the training distribution without generating additional images, thereby avoiding introducing spurious artifacts. Additionally, interpolation toward real-image representations further enables the construction of hard fake features in close proximity to real features, providing challenging examples to support learning a more discriminative real–fake decision boundary. Through supervised learning on these real and fake data, we obtain an effective AIGI detector.

As training progresses and the detector improves, a natural question arises: can the detector guide the synthesis of more informative fake features that, in turn, improve its own performance? Inspired by adversarial learning Goodfellow et al. (2014), we turn this idea into a framework that alternates between detector training and fake synthesis. Specifically, we first fix the synthesis mechanism and train the detector to correctly classify real and fake features, including the synthesized ones. We then fix the detector and train the synthesis mechanism to maximize the detector's classification loss on fake-labeled synthesized features. This encourages the synthesis of fake features that the current detector finds difficult to recognize. By alternating these updates, the detector guides fake features synthesis, and the resulting challenging fake features help improve the detector.

Another key question is which component of the synthesis mechanism to update during adversarial training. Updating the pretrained encoder would increase substantial computational costs and disrupt its powerful representations. We therefore turn to the mixing coefficient. However, existing mixing strategies often use a fixed coefficient. Differences in feature separation and classification difficulty across real–fake pairs motivate pair-adaptive coefficients that balance sample difficulty and fake-label reliability. To enable both adaptive mixing and detector-guided synthesis, we introduce a lightweight mixing coefficient (LMC) generator. Given a pair of real and fake features, it generates a pair-specific coefficient for feature interpolation. During adversarial updates, the LMC generator is optimized using detector feedback. However, we observe coefficient collapse as adversarial training progresses: the generator produces nearly identical coefficients across different real–fake pairs, reducing its adaptivity. To address this issue, we introduce a coefficient diversity constraint to mitigate collapse and preserve variation across pairs. Extensive experiments demonstrate the effectiveness of our method and its generalization to unseen generators and image domains.

Our main contributions are summarized as follows:

- We propose a feature-space synthesis strategy for aligned real–fake pairs, which constructs fake-labeled training features by interpolating their representations rather than manipulating images in pixel space. This avoids introducing additional pixel-level artifacts while enabling the generation of challenging fake features close to real representations.

- We formulate detector learning and fake-feature synthesis as an alternating adversarial optimization process, where the evolving detector provides feedback to guide the synthesis of informative fake features, and these challenging features are subsequently used to further improve the detector.

- We introduce a lightweight mixing coefficient (LMC) generator to realize pair-adaptive synthesis by predicting a bounded interpolation coefficient for each real–fake pair. To prevent coefficient collapse during adversarial training, we further impose a diversity constraint that preserves variation of coefficient across pairs.

- We conduct extensive experiments on nine benchmarks covering diverse generators and image domains. The results demonstrate improved generalization to unseen generators and datasets, validating the effectiveness of feature-space synthesis and detector-guided fake synthesis enhancement.

2 Related Work

2.1 Generalizable AI-Generated Image Detection

Early learning-based detectors identify systematic spatial artifacts shared by CNN-based generators Wang et al. (2020), while frequency-based methods exploit spectral anomalies introduced by generative architectures Frank et al. (2020). Subsequent work investigates more transferable upsampling-related patterns Tan et al. (2024) and reconstruction discrepancies produced by diffusion models Wang et al. (2023); Ricker et al. (2024). Another line of research leverages the broad representations of pretrained vision foundation models. For example, UniversalFakeDetect uses frozen CLIP features for cross-generator detection Ojha et al. (2023), while later work further adapts CLIP representations to improve AIGI detection Cozzolino et al. (2024). Effort Yan et al. (2025b) uses SVD to construct orthogonal semantic and forgery subspaces, preserving pretrained semantic knowledge while learning forgery cues through residual components. Beyond representation adaptation, BPL Xu et al. (2026) employs bias-only pair-aware learning to emphasize generation-induced discrepancies between real and fake samples. Despite these advances, detectors can still associate authenticity with generator-specific artifacts or dataset-dependent semantics, limiting their transfer to unseen generators and image domains Zheng et al. (2024).

2.2 Data Alignment and Pair-Aware Learning

Data alignment reduces shortcut learning by making the real and fake training distributions more comparable. AlignedForensics Rajan et al. (2025) reconstructs real images using a latent diffusion autoencoder, producing real–fake pairs with closely matched content, resolution, and format. DRCT Chen et al. (2024) constructs hard samples through diffusion reconstruction and combines classification with contrastive learning to guide the detector toward subtle generation artifacts. DDA Chen et al. (2025) further addresses residual discrepancies through joint pixel- and frequency-domain alignment. These studies highlight the value of reconstruction-based data construction and alignment for generalizable detector learning. Our work uses aligned real-fake pairs to synthesize fake features in feature space, avoiding artifacts associated with pixel-level operation. Through alternating optimization, a lightweight operator uses detector feedback to predict bounded, pair-adaptive mixing coefficients, while a diversity constraint discourages coefficient collapse.

3 Methodology

3.1 Problem formulation and notation

Given only image-level ground-truth labels, AIGI detectors must learn the distinction between real and synthetic images from the training data. However, the real and synthetic subsets may differ systematically in attributes unrelated to the image-generation process, such as semantic content, image resolution, encoding format, and compression statistics. Since these spurious cues are often more salient than subtle generation-related forensic traces, detectors may associate them with authenticity and learn dataset-specific decision boundaries, resulting in poor generalization to unseen generators and data domains.

To reduce the influence of such generation-irrelevant discrepancies, following Rajan et al. (2025); Chen et al. (2025) we adopt an aligned-pair training setting and denote the training set as

$$\mathcal{P} = \left\{ \Big((x_i^r, y_i^r), (x_i^f, y_i^f) \Big) \right\}_{i=1}^{N}, \tag{1}$$

where x_i^r denotes the i-th real image and x_i^f denotes its aligned synthetic counterpart, while y_i^r and y_i^f denote their real and fake labels, respectively. Here, N is the total number of aligned pairs, and each mini-batch contains B such pairs.

We formulate the detector as $F_{\theta,\psi} = C_\psi \circ E_\theta$, where E_θ is a shared forensic encoder and C_ψ is a binary classifier followed by a sigmoid function. We implement E_θ with a pretrained DINOv2 Oquab et al. (2024). For the i-th aligned pair, the encoder extracts the corresponding features as

$$\boldsymbol{z}_i^r = E_\theta(x_i^r), \qquad \boldsymbol{z}_i^f = E_\theta(x_i^f). \tag{2}$$

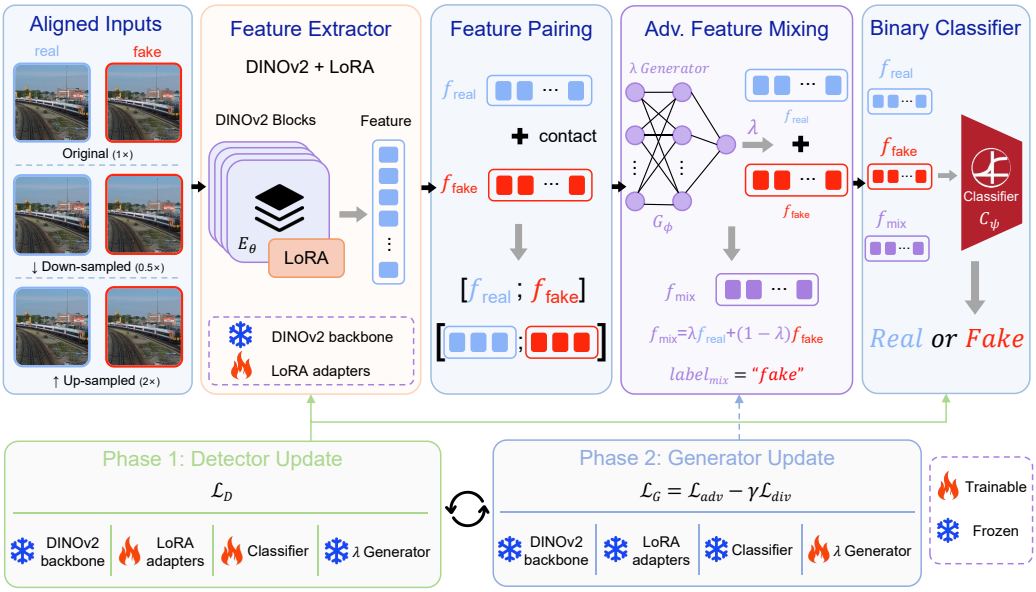


Figure 1: Overview of the proposed framework. Given aligned real-fake pairs, the lightweight mixing coefficient (LMC) generator G_ϕ generates pair-adaptive interpolation coefficients to synthesize fake-labeled hard features. The detector and G_ϕ are optimized alternately.

These paired features serve as the basis for subsequent feature synthesis. Our goal is to learn generation-related forensic cues that generalize across unseen generators and data domains. Fig. 1 provides an overview of our framework, which extracts aligned features, synthesizes pair-adaptive hard features, and alternately optimizes the detector and the lightweight mixing coefficient (LMC) generator G_ϕ. Details of the optimization procedure are presented in Appendix B.

3.2 Feature-Space Fake Synthesis for Detector Learning

Real and synthetic training images often differ in generation-irrelevant attributes, which may be exploited as shortcuts and hinder generalization Zheng et al. (2024); Rajan et al. (2025). Recent data-alignment methods reduce such discrepancies by constructing more closely matched real-fake pairs in pixel space Rajan et al. (2025); Chen et al. (2024; 2025). However, pixel-space fake synthesis may still introduce operation-specific artifacts that themselves become shortcut cues.

Given a real image x^r, AlignedForensics Rajan et al. (2025) first obtains its VAE reconstruction x^{rec} and uses the reconstructed image as a pixel-level synthetic counterpart for detector training. However, DDA Chen et al. (2025) shows that reconstruction alone does not fully eliminate shortcut cues, leaving frequency-level discrepancies, particularly in high-frequency details, as well as residual pixel-level misalignment. To address these issues, DDA further performs high-frequency fusion to reduce frequency-domain discrepancies and applies pixel-level mixup between real and reconstructed images to improve pixel-space alignment by. Nevertheless, these additional pixel-space operations may themselves introduce operation-specific artifacts, such as pixel-level ghosting, which can become new shortcut cues.

Motivated by the above observation, we move fake-sample construction from pixel space to feature space. Rather than generating additional fake images through pixel-level transformations, we directly synthesize fake-labeled representations from aligned real–fake pairs. This preserves the benefit of data alignment while avoiding additional image-level operations that may introduce new shortcut cues. Moreover, interpolating fake representations toward their real counterparts enables the construction of more challenging fake samples, providing stronger supervision for detector learning.

Specifically, let $\boldsymbol{z}_i^r$ and $\boldsymbol{z}_i^f$ denote the encoded representations of an aligned real–fake pair. We construct a synthetic fake representation through linear interpolation:

$$\boldsymbol{z}_i^m = \lambda_0 \boldsymbol{z}_i^r + (1 - \lambda_0)\boldsymbol{z}_i^f, \tag{3}$$

where $\lambda_0 \in (0, 1)$ controls the contribution of the real representation. The synthesized representation $\boldsymbol{z}_i^m$ is assigned the fake label. As λ_0 increases, $\boldsymbol{z}_i^m$ moves closer to its real counterpart, making the synthesized fake representation more difficult to distinguish from real ones. In this way, feature-space synthesis not only avoids additional pixel-level artifacts but also enriches the training distribution with challenging fake representations that encourage the detector to learn a more discriminative real–fake boundary. We first consider a fixed λ_0 shared across all aligned pairs; its adaptive extension is introduced in the following section.

Detector training and inference. Given the real, fake, and synthesized representations, the binary classifier C_ψ followed by a sigmoid function predicts their fake probabilities as

$$p_i^r = C_\psi(\boldsymbol{z}_i^r), \qquad p_i^f = C_\psi(\boldsymbol{z}_i^f), \qquad p_i^m = C_\psi(\boldsymbol{z}_i^m). \tag{4}$$

We treat $\boldsymbol{z}_i^r$ as a real sample and both $\boldsymbol{z}_i^f$ and $\boldsymbol{z}_i^m$ as fake samples, and optimize the detector using

$$\mathcal{L}_D = -\frac{1}{3B} \sum_{i=1}^{B} \left[\log(1 - p_i^r) + \log p_i^f + \log p_i^m \right]. \tag{5}$$

This objective jointly learns from the original real–fake pairs and the synthesized fake representations, encouraging the detector to distinguish fake samples even when their representations are close to those of real images. Similar to Chen et al. (2025), we minimize $\mathcal{L}_D$ to optimize encoder E_θ and classifier C_ψ.

At inference, given a test image x, its fake probability is computed as

$$p = C_\psi(E_\theta(x)), \tag{6}$$

and the image is classified as fake if $p \geq 0.5$ and real otherwise.

3.3 Adversarial Enhancement of Synthesized Fake Features

As training progresses and the detector becomes more discriminative, a natural question arises: can the evolving detector guide the synthesis of more informative fake features that, in turn, further improve its own performance? Inspired by adversarial learning Goodfellow et al. (2014); Madry et al. (2019), we formulate detector learning and fake-feature synthesis as an alternating adversarial optimization process. Specifically, we alternate between two stages: (1) during the detector-update stage, the synthesis mechanism is fixed and the detector is optimized to correctly classify real features, original fake features, and synthesized fake features; (2) during the synthesis-update stage, the detector is fixed and the synthesis mechanism is optimized to increase the detector's classification loss on synthesized fake features, thereby encouraging the generation of fake-labeled features that are more difficult for the current detector to recognize. These challenging features are then used in the subsequent detector-update stage. Through this alternating process, feature synthesis continuously adapts to the evolving detector, while the detector is repeatedly trained on informative hard examples, forming a closed-loop interaction between synthesis and detector learning.

A key design question is which component of the synthesis mechanism should be adapted during adversarial optimization. Directly updating the pretrained encoder would incur substantial computational and memory costs and may disrupt its learned representations. We therefore adapt the mixing coefficient, which provides a lightweight means of controlling the location of the synthesized feature along the real–fake interpolation path. A single shared coefficient, however, cannot account for variations in feature separation and classification difficulty across aligned real–fake pairs. The same interpolation strength may provide little additional supervision for some pairs while introducing excessive label ambiguity for others. To enable pair-adaptive synthesis under detector guidance, we introduce a lightweight mixing coefficient (LMC) generator G_ϕ that predicts an interpolation coefficient λ_i for each aligned feature pair.

Specifically, we implement G_ϕ as a lightweight multilayer perceptron composed of several fully connected layers. For the i-th aligned real–fake pair, the generator takes the concatenated endpoint

representations $[\boldsymbol{z}_i^r; \boldsymbol{z}_i^f]$ as input and outputs a scalar that determines the interpolation strength. We first map this output to $(0, 1)$ using a sigmoid function and then rescale it to a predefined interval $[\lambda_{\min}, \lambda_{\max}]$:

$$\begin{aligned} \widetilde{\lambda}_i &= \sigma\left(G_\phi\left([\boldsymbol{z}_i^r; \boldsymbol{z}_i^f]\right)\right), \\ \lambda_i &= \lambda_{\min} + (\lambda_{\max} - \lambda_{\min})\,\widetilde{\lambda}_i, \end{aligned} \tag{7}$$

where $[\cdot;\cdot]$ denotes feature concatenation and $\sigma(\cdot)$ is the sigmoid function.

The resulting coefficient is then used to synthesize a fake feature:

$$\boldsymbol{z}_i^m = \lambda_i \boldsymbol{z}_i^r + (1 - \lambda_i)\boldsymbol{z}_i^f. \tag{8}$$

λ_i determines the interpolation position, with larger values moving $\boldsymbol{z}_i^m$ closer to the real endpoint and generally increasing the difficulty of fake classification. The interval $[\lambda_{\min}, \lambda_{\max}]$ restricts synthesis to a controlled portion of the interpolation path: the lower bound enforces a minimum contribution from the real representation, whereas the upper bound prevents excessive movement toward the real endpoint and thereby reduces the risk of label ambiguity. Conditioning G_ϕ on both endpoint features allows the interpolation position to adapt to pair-specific differences in feature geometry and classification difficulty.

During the generator-update stage, the endpoint features are detached from the computation graph and the detector parameters are fixed, such that gradients are propagated only to G_ϕ. Since each synthesized feature is assigned the fake label, we define the adversarial objective as

$$\mathcal{L}_{\text{adv}} = \frac{1}{B}\sum_{i=1}^{B} \log p_i^m, \tag{9}$$

where p_i^m denotes the detector's predicted fake-class probability for $\boldsymbol{z}_i^m$. Minimizing $\mathcal{L}_{\text{adv}}$ decreases p_i^m, encouraging G_ϕ to select pair-specific coefficients that produce fake features that are more difficult for the fixed detector to recognize. As the detector evolves across alternating updates, its feedback continuously reshapes the coefficient generator, allowing synthesis to adapt jointly to pair-specific characteristics and the detector's current decision behavior.

3.4 Preventing Collapse of Coefficient Generator

Despite pair-conditioned prediction, adversarial optimization may still drive the coefficients toward a common high-value solution. Since increasing λ_i moves a mixed feature closer to its real endpoint and generally makes it more difficult to classify as fake, G_ϕ may produce coefficients concentrated near $\lambda_{\max}$. As shown in Fig. 2(a), without diversity regularization, the mean coefficient approaches its upper bound while the coefficient standard deviation decreases toward zero. This behavior weakens pair adaptivity by assigning different pairs similar relative mixing positions, thereby limiting the diversity of hard negatives explored during training.

To counter this tendency, we encourage coefficient diversity within each mini-batch by computing the sample standard deviation of the B predicted coefficients:

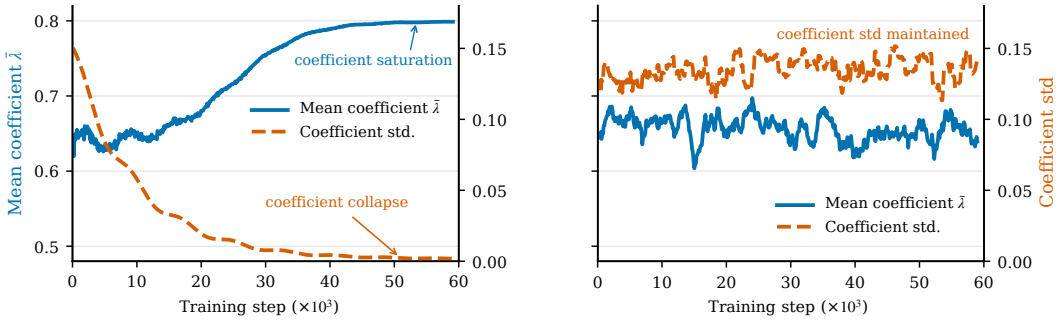
$$\mathcal{L}_{\text{div}} = \sqrt{\frac{\sum_{i=1}^{B}(\lambda_i - \bar{\lambda})^2}{B-1}}, \quad \text{where } \bar{\lambda} = \frac{1}{B}\sum_{i=1}^{B}\lambda_i. \tag{10}$$

The final objective for G_ϕ is defined as

$$\mathcal{L}_G = \mathcal{L}_{\text{adv}} - \gamma\mathcal{L}_{\text{div}}, \tag{11}$$

where $\gamma > 0$ controls the diversity regularization. Because $\mathcal{L}_G$ is minimized, the negative diversity term discourages the predicted coefficients from collapsing to a common value within the prescribed range. As shown in Fig. 2(b), coefficient diversity is maintained over the recorded training steps, indicating that the predicted coefficients retain diversity across pairs.

Alternating optimization. Training alternates between two phases. We first fix G_ϕ and update the detector using Eq. 5. We then fix the detector and update G_ϕ using Eq. 11, allowing detector feedback to guide subsequent feature synthesis. The complete procedure is provided in Appendix B.



(a) Without $\mathcal{L}_{\text{div}}$ (b) With $\mathcal{L}_{\text{div}}$

Figure 2: Evolution of the mean mixing coefficient and coefficient diversity during training. (a) Without diversity regularization, the coefficients concentrate near the upper bound as their standard deviation decreases. (b) With diversity regularization, non-zero coefficient diversity is maintained.

4 Experiments

4.1 Experimental Setup

Datasets. Following DDA Chen et al. (2025), all compared detectors are evaluated on nine diverse benchmarks, including six manually curated datasets: GenImage Zhu et al. (2023), DRCT-2M Chen et al. (2024), EvalGEN Chen et al. (2025), Synthbuster Bammey (2024), ForenSynths Wang et al. (2020), and AIGCDetectionBenchmark Zhong et al. (2024); and three in-the-wild datasets: Chameleon Yan et al. (2025a), WildRF Cavia et al. (2024), and SynthWildX Cozzolino et al. (2024). These benchmarks cover images from diverse sources and generators, including GANs, diffusion models, autoregressive models, and unknown online generators.

Implementation Details. We utilize DINOv2 Oquab et al. (2024) as the backbone and fine-tune it using LoRA Hu et al. (2022) with a rank of 8. The input size is set to 336×336. Training alternates between updating the detector and the lightweight coefficient-prediction network G_ϕ, using learning rates of 1×10^{-4} and 1×10^{-5}, respectively. The model is trained for one epoch with a batch size of 16. The training data consist exclusively of MSCOCO Lin et al. (2014) images and their DDA-aligned counterparts. A single trained model is evaluated on all benchmarks without dataset-specific fine-tuning.

Evaluation Metrics and Comparative Methods. We report ACC, AUC, and AP. The main paper focuses on ACC, while complete AUC and AP results are provided in the appendix. We compare our method with nine representative detectors: NPR Tan et al. (2024), UnivFD Ojha et al. (2023), FatFormer Liu et al. (2024), SAFE Li et al. (2025), C2P-CLIP Tan et al. (2025), AIDE Yan et al. (2025a), DRCT Chen et al. (2024), AlignedForensics Rajan et al. (2025), and DDA Chen et al. (2025). For fairness and reproducibility, we use their officially released checkpoints and evaluate all methods under the same protocol.

4.2 Comparison with Existing Methods

Overall performance across benchmarks. Table 1 compares detection accuracy across nine benchmarks. Most baselines achieve mean accuracies below 75%, and the strongest baseline reaches 79.69%. Our method achieves the highest mean accuracy of 89.14%, an improvement of 9.45 percentage points over the strongest baseline. It also outperforms that baseline on every benchmark and ranks first on eight of the nine benchmarks; on the remaining benchmark, it trails the best result by only 0.17 percentage points. The improvements span both curated and in-the-wild datasets, suggesting that the gain is not confined to a particular evaluation setting. These results support the effectiveness of detector-guided feature synthesis for generalization to unseen generators and image domains. Detailed results for the subsets of all nine benchmarks are provided in Appendix C.

Table 1: Average detection accuracy (%) across nine benchmarks. Mean is the unweighted average of the benchmark-level results. AIGCDB denotes AIGCDetectionBenchmark.

Method	AIGCDB	DRCT-2M	WildRF	ForenSynths	EvalGEN	Synthbuster	SynthWildX	Chameleon	GenImage	Mean
NPR	50.54	59.06	55.12	41.92	30.85	39.17	50.27	59.44	53.05	48.83
UnivFD	72.49	62.01	51.66	77.68	41.47	55.30	52.23	57.39	64.05	59.36
FatFormer	74.78	51.95	55.60	84.52	48.93	53.62	53.38	58.03	61.03	60.20
SAFE	50.05	59.31	54.18	40.71	31.84	52.76	49.93	59.19	51.14	49.90
C2P-CLIP	70.15	55.55	54.57	87.01	40.61	68.06	53.15	57.59	62.09	60.98
AIDE	53.05	64.27	52.07	40.07	59.77	50.90	53.16	59.97	63.82	55.32
DRCT	81.40	90.44	54.91	73.83	82.77	61.87	55.44	51.63	88.19	71.17
AlignedForensics	65.95	95.30	82.41	51.77	67.31	78.23	78.97	70.20	77.60	74.19
DDA	86.92	85.18	78.14	77.91	73.74	84.15	75.11	73.03	83.01	79.69
Ours	**92.52**	**96.21**	**89.37**	**86.84**	**89.76**	**90.97**	**88.77**	**74.24**	**93.56**	**89.14**

Table 2: Generator-wise ACC (%) results on GenImage and SynthWildX.

Method	GenImage									SynthWildX			
	ADM	Big-GAN	GLIDE	Mid-journey	SD1.4	SD1.5	VQDM	Wukong	Avg.	DALL-E3	Firefly	Mid-journey	Avg.
NPR	53.24	59.80	51.19	48.78	51.09	50.93	56.48	52.93	53.05	51.15	50.00	49.65	50.27
UnivFD	62.51	84.37	61.28	55.06	55.58	55.67	76.88	61.10	64.05	45.55	65.45	45.69	52.23
FatFormer	59.83	81.77	60.28	51.99	52.81	52.76	69.98	58.86	61.03	48.11	61.79	50.25	53.38
SAFE	50.41	51.84	53.73	49.89	50.55	50.54	51.01	51.15	51.14	49.95	50.41	49.44	49.93
C2P-CLIP	53.61	92.69	65.89	50.60	55.03	55.33	63.54	60.03	62.09	50.26	59.76	49.44	53.15
AIDE	65.68	81.70	81.91	51.28	58.58	74.46	50.08	46.85	63.82	47.76	46.00	65.71	53.16
DRCT	79.81	89.51	89.05	85.48	91.36	90.88	**88.66**	90.79	88.19	58.75	56.71	50.86	55.44
AlignedForensics	50.63	51.45	53.30	**96.12**	**99.88**	**99.81**	69.81	**99.83**	77.60	87.00	54.67	**95.24**	78.97
DDA	80.74	86.49	88.50	76.33	86.78	86.46	70.62	88.15	83.01	80.45	68.70	76.19	75.11
Ours	**92.46**	**95.46**	**94.56**	85.52	97.60	97.62	87.83	97.43	**93.56**	**90.79**	**85.26**	90.27	**88.77**

Generator-wise performance on GenImage and SynthWildX. Table 2 reports results for individual generators on these two benchmarks. Our method achieves the highest average accuracy on both GenImage (93.56%) and SynthWildX (88.77%), exceeding the strongest baseline on each by 5.37 and 9.80 percentage points, respectively. Compared with DDA, it improves accuracy on all 11 generator subsets. The gains are particularly pronounced on GenImage's VQDM (87.83% versus 70.62%) and SynthWildX's Firefly (85.26% versus 68.70%). AlignedForensics achieves higher accuracy on the Midjourney subsets of both benchmarks, indicating that performance still varies by generator.

Table 3: Component ablation across nine AIGI detection benchmarks. Results are reported in ACC (%). AIGCDB denotes AIGCDetectionBenchmark.

Feature Synthesis	Adversarial Opt.	Diversity Reg.	AIGCDB	DRCT-2M	WildRF	Foren-Synths	EvalGEN	Synth-buster	Synth-WildX	Chameleon	GenImage	Avg.
×	×	×	92.47	92.98	81.85	86.16	66.67	87.45	78.29	66.67	91.10	82.64
✓	×	×	90.86	95.99	86.26	78.72	89.46	88.32	86.47	68.01	90.80	86.10
✓	✓	×	91.52	95.91	88.83	81.69	84.41	90.51	87.08	69.28	92.53	86.86
✓	✓	✓	**92.52**	**96.21**	**89.37**	**86.84**	**89.76**	**90.97**	**88.77**	**74.24**	**93.56**	**89.14**

4.3 Ablation Studies

We conduct ablation studies to evaluate feature-space synthesis, detector-guided adversarial optimization, and coefficient-diversity regularization. Table 3 reports the mean accuracy across nine benchmarks. The detector without feature-space synthesis achieves an average accuracy of 82.64%. Adding feature-space synthesis raises it to 86.10%, a gain of 3.46 percentage points. Incorporating adversarial optimization further improves the average to 86.86%. With coefficient-diversity regularization, the full method reaches 89.14%, improving by 2.28 percentage points over the variant without this constraint and by 6.50 percentage points over the baseline without feature-space synthe-

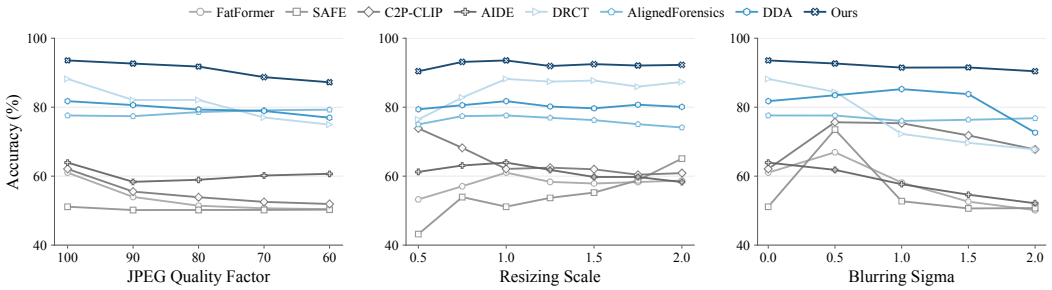


Figure 4: Robustness analysis to image post-processing on GenImage.

sis. The full method also achieves the highest accuracy on all nine benchmarks among the evaluated variants.

4.4 Hyperparameter Sensitivity

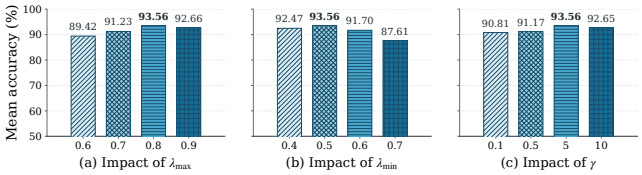


Figure 3: Hyperparameter sensitivity on GenImage.

We evaluate hyperparameter sensitivity using the average accuracy on GenImage, varying one parameter at a time. The reference configuration used in the main experiments is $\lambda_{\min} = 0.5$, $\lambda_{\max} = 0.8$, and $\gamma = 5$, which achieves 93.56%. As shown in Fig. 3(a), with $\lambda_{\min}$ fixed at 0.5, increasing $\lambda_{\max}$ from 0.6 to 0.8 improves accuracy from 89.42% to 93.56%, followed by a slight decrease to 92.66% at 0.9. With $\lambda_{\max}$ fixed at 0.8, Fig. 3(b) shows the best result at $\lambda_{\min} = 0.5$; increasing it to 0.7 reduces accuracy to 87.61%. Finally, with both bounds fixed, Fig. 3(c) shows that accuracy rises from 90.81% at $\gamma = 0.1$ to 93.56% at $\gamma = 5$, then decreases to 92.65% at $\gamma = 10$. The reference configuration achieves the highest accuracy among the tested settings in all three analyses.

4.5 Robustness Analysis to Common Image Post-processing

Following DDA, we evaluate robustness on GenImage-JPEG96 under JPEG compression, resizing, and Gaussian blur. Specifically, we consider JPEG quality factors from 100 to 60, resizing scales from 0.5 to 2.0, and blur strengths from 0 to 2.0. Each operation is applied equally to real and fake images, while the checkpoint and decision threshold remain fixed. As shown in Fig. 4, our method maintains consistently high performance across different degradation levels, achieving an average accuracy of 91.7%. The lowest accuracy is 87.23% under JPEG compression with a quality factor of 60, corresponding to a drop of only 6.33 percentage points from the clean accuracy of 93.56%. These results indicate that feature-space hard-negative learning reduces reliance on fragile pixel-level artifacts.

5 Conclusion

We presented a framework for generalizable AI-generated image detection that synthesizes fake features from aligned real-fake pairs. By mixing their representations after feature extraction, the method avoids the pixel-space operation that may introduce classification shortcuts. During alternating optimization, detector feedback guides a lightweight generator to predict bounded, pair-adaptive mixing coefficients, while a diversity constraint discourages coefficient collapse. The generator is used only during training, so inference requires only the encoder and classifier. Experiments across nine benchmarks show improved generalization to unseen generators and image domains.

AI USE STATEMENT

We used generative AI tools only to improve the clarity, grammar, and readability of the manuscript. We developed, checked, and approved all technical content, including the research motivation, method design, experimental protocol, interpretation of results, and final claims. We remain fully responsible for the accuracy and integrity of the paper.

ETHICS STATEMENT

We study AI-generated image detection using existing image benchmarks and aligned synthetic counterparts. We did not recruit human participants, and we do not attempt to identify individuals or infer sensitive attributes from the images. We acknowledge the original data sources and encourage any reuse or redistribution of benchmark images to respect their licenses, access conditions, and applicable privacy requirements. We recognize that detection errors may occur, particularly for images from unseen generators or those subjected to post-processing. We therefore regard model predictions as supporting evidence rather than conclusive proof of authenticity in consequential settings. We report our experimental settings and limitations to support responsible interpretation of the results.

REPRODUCIBILITY STATEMENT

We specify the aligned-pair setting, feature-synthesis formulation, training objectives, and alternating update procedure in the method section and Algorithm 1. The experimental section describes the benchmarks, evaluation protocol, and implementation settings used in our comparisons. We report component ablations and hyperparameter sensitivity in the main text, and provide subset-level results in Appendix C. These materials document the procedure and results needed to examine and reproduce our findings.

REFERENCES

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.

Bar Cavia, Eliahu Horwitz, Tal Reiss, and Yedid Hoshen. Real-time deepfake detection in the real-world, 2024. URL https://arxiv.org/abs/2406.09398.

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. URL https://openreview.net/forum?id=oRLwyayrh1.

Ruoxin Chen, Junwei Xi, Zhiyuan Yan, Ke-Yue Zhang, Shuang Wu, Jingyi Xie, Xu Chen, Lei Xu, Isabel Guan, Taiping Yao, and Shouhong Ding. Dual data alignment makes AI-generated image detector easier generalizable. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=C39ShJwtD5.

Davide Cozzolino, Giovanni Poggi, Riccardo Corvi, Matthias Nießner, and Luisa Verdoliva. Raising the bar of ai-generated image detection with clip. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pp. 4356–4366, June 2024.

Joel Frank, Thorsten Eisenhofer, Lea Schönherr, Asja Fischer, Dorothea Kolossa, and Thorsten Holz. Leveraging frequency analysis for deep fake image recognition. In *ICML*, pp. 3247–3258, 2020. URL http://proceedings.mlr.press/v119/frank20a.html.

Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014. URL https://arxiv.org/abs/1406.2661.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Ouxiang Li, Jiayin Cai, Yanbin Hao, Xiaolong Jiang, Yao Hu, and Fuli Feng. Improving synthetic image detection towards generalization: An image transformation perspective. In *Proceedings of the 31st ACM SIGKDD conference on knowledge discovery and data mining v. 1*, pp. 2405–2414, 2025.

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 *European conference on computer vision*, pp. 740–755. Springer, 2014.

Huan Liu, Zichang Tan, Chuangchuang Tan, Yunchao Wei, Jingdong Wang, and Yao Zhao. Forgery-aware adaptive transformer for generalizable synthetic image detection. In *CVPR*, pp. 10770–10780, 2024. URL https://doi.org/10.1109/CVPR52733.2024.01024.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks, 2019. URL https://arxiv.org/abs/1706.06083.

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, June 2023.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel HAZIZA, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=a68SUt6zFt. Featured Certification.

Anirudh Sundara Rajan, Utkarsh Ojha, Jedidiah Schloesser, and Yong Jae Lee. Aligned datasets improve detection of latent diffusion-generated images. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=doBkiqESYq.

Jonas Ricker, Denis Lukovnikov, and Asja Fischer. Aeroblade: Training-free detection of latent diffusion images using autoencoder reconstruction error. In *CVPR*, pp. 9130–9140, 2024. URL https://doi.org/10.1109/CVPR52733.2024.00872.

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

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, June 2024.

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*, volume 39, pp. 7184–7192, 2025.

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)*, June 2020.

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

Yuxiang Xu, Rundong He, Zhiyuan Yan, Yicong Dong, Zhongyi Han, Xiaoyan Wang, and Yilong Yin. BPL: Generalizable deepfake detection via bias-only pair-aware learning. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=oUWm8DaxjJ.

Shilin Yan, Ouxiang Li, Jiayin Cai, Yanbin Hao, Xiaolong Jiang, Yao Hu, and Weidi Xie. A sanity check for AI-generated image detection. In *The Thirteenth International Conference on Learning Representations*, 2025a. URL https://openreview.net/forum?id=ODRHZrkOQM.

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 *Forty-second International Conference on Machine Learning*, 2025b. URL https://openreview.net/forum?id=GFpjO8S8Po.

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 A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 59570–59596. Curran Associates, Inc., 2024. doi: 10.52202/079017-1903. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/6dddcff5b115b40c998a08fbd1cea4d7-Paper-Conference.pdf.

Nan Zhong, Yiran Xu, Sheng Li, Zhenxing Qian, and Xinpeng Zhang. Patchcraft: Exploring texture patch for efficient ai-generated image detection, 2024. URL https://arxiv.org/abs/2311.12397.

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 *NeurIPS*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/f4d4a021f9051a6c18183b059117e8b5-Abstract-Datasets_and_Benchmarks.html.

A Potential Shortcuts in Pixel-Space Interpolation

We train a detector on real images and pixel-mixed fake samples to examine whether pixel-space interpolation introduces shortcut cues. Given a real image x^r, its VAE reconstruction x^{rec}, and a mixing coefficient λ denoting the real-image contribution, we construct a PixelMix sample as

$$x^p_\lambda = \lambda x^r + (1-\lambda)x^{\mathrm{rec}}. \tag{12}$$

Table 4: Responses to Recon and PixelMix samples.

Sample	λ	Fake Detection Rate (%)	Mean Fake Score (%)	Score > Recon (%)
Recon	–	63.6	65.2	–
PixelMix	0.2	92.4	90.1	99.6
	0.3	95.2	92.8	99.5
	0.4	95.0	92.5	98.0
	0.5	90.9	88.4	92.8
	0.6	80.5	77.8	73.5

We compare paired Recon and PixelMix samples derived from the same source images. As shown in Table 4, PixelMix produces substantially stronger fake responses across all mixing ratios, with most samples receiving higher fake scores than their corresponding Recon. Since the paired samples mainly differ in the interpolation operation, these results indicate that pixel-space interpolation can introduce shortcut cues.

B Alternating Optimization Procedure

Algorithm 1 details the updates performed for each mini-batch. After the detector update, we recompute the endpoint features with the updated encoder and detach them before optimizing the coefficient generator. Only the trained encoder and classifier are retained for inference.

Algorithm 1 Alternating optimization of the detector and coefficient generator

Input: Aligned training pairs $\mathcal{P}$; encoder E_θ; classifier C_ψ; coefficient generator G_ϕ; coefficient bounds $\lambda_{\min}$ and $\lambda_{\max}$; regularization weight γ
Output: Trained encoder E_θ and classifier C_ψ

for each mini-batch of B aligned pairs from $\mathcal{P}$ **do**
  ▷ **Detector phase**
  Fix G_ϕ
  Compute $\boldsymbol{z}^r_i = E_\theta(x^r_i)$ and $\boldsymbol{z}^f_i = E_\theta(x^f_i)$
  Compute λ_i using Eq. 7 and detach λ_i
  Construct $\boldsymbol{z}^m_i = \lambda_i \boldsymbol{z}^r_i + (1-\lambda_i)\boldsymbol{z}^f_i$
  Compute p^r_i, p^f_i, and p^m_i using C_ψ
  Update θ and ψ by minimizing $\mathcal{L}_D$ in Eq. 5
  ▷ **Generator phase**
  Fix E_θ and C_ψ
  Recompute $\boldsymbol{z}^r_i$ and $\boldsymbol{z}^f_i$ with E_θ and detach both features
  Compute λ_i using Eq. 7
  Construct $\boldsymbol{z}^m_i = \lambda_i \boldsymbol{z}^r_i + (1-\lambda_i)\boldsymbol{z}^f_i$
  Compute $p^m_i = C_\psi(\boldsymbol{z}^m_i)$
  Compute $\mathcal{L}_{\mathrm{adv}}$ and $\mathcal{L}_{\mathrm{div}}$ using Eqs. 9 and 10
  Update ϕ by minimizing $\mathcal{L}_G$ in Eq. 11
end for
return E_θ and C_ψ

C Detailed Experimental Results

C.1 Detailed ACC Results

We report detailed ACC results on the seven benchmarks not expanded in the main paper. For benchmarks containing multiple generators or data sources, we report the results on each subset together with the macro average. Detailed results on GenImage and SynthWildX are provided in the main paper.

Table 5: Detailed ACC (%) results on AIGCDetectionBenchmark. WFIR denotes WhichFaceIsReal.

Method	ADM	Big-GAN	Cycle-GAN	DALL-E2	Gau-GAN	GLIDE	Mid-journey	Pro-GAN	SDXL	SD1.4	SD1.5	Star-GAN	Style-GAN	Style-GAN2	VQDM	WFIR	Wu-kong	Avg.
NPR	53.24	44.95	62.38	50.75	35.96	51.19	48.78	51.03	49.38	51.09	50.93	49.90	50.31	50.14	56.48	49.80	52.93	50.54
UnivFD	62.51	87.50	96.74	49.95	98.79	61.28	55.06	99.38	58.15	55.58	55.67	95.12	79.99	69.41	76.88	69.20	61.10	72.49
FatFormer	59.83	97.90	99.28	49.00	99.17	60.28	51.99	99.21	63.75	52.81	52.76	99.75	87.51	81.63	69.98	87.50	58.86	74.78
SAFE	50.41	48.95	52.99	50.05	42.60	53.73	49.89	49.96	49.93	50.55	50.54	49.95	49.99	49.99	51.01	49.10	51.15	50.05
C2P-CLIP	53.61	92.00	99.32	50.20	98.20	65.89	50.60	97.06	55.03	55.33	51.20	99.70	76.96	58.51	63.54	65.40	60.03	70.15
AIDE	51.28	54.65	58.58	65.68	50.08	43.80	46.97	40.68	41.91	52.43	81.70	81.91	38.92	40.85	37.17	40.70	74.46	53.05
DRCT	79.81	81.35	90.73	89.30	93.84	89.05	85.48	71.05	91.36	90.88	88.20	53.00	62.74	63.80	88.66	73.85	90.79	81.40
AlignedForensics	50.63	50.60	49.17	50.70	50.21	53.30	96.12	50.54	99.88	99.81	96.33	51.83	51.89	50.63	69.81	50.00	99.83	65.95
DDA	89.68	87.42	82.88	92.90	83.69	89.72	86.33	86.25	92.58	91.76	92.03	62.03	85.43	85.43	85.63	91.80	92.09	86.92
Ours	92.46	94.70	87.94	98.20	95.64	94.56	85.52	94.50	98.62	97.60	97.62	65.03	95.37	95.22	87.83	94.55	97.43	92.52

AIGCDetectionBenchmark. Table 5 reports accuracy on 17 subsets. Our method achieves the highest average accuracy of 92.52%, exceeding the strongest baseline, DDA, by 5.60 percentage points. It outperforms DDA on 16 of the 17 subsets, with gains of 11.95, 9.94, and 9.79 points on GauGAN, StyleGAN, and StyleGAN2, respectively. Midjourney is the only subset where our accuracy is slightly lower than DDA's (85.52% versus 86.33%).

Table 6: Detailed ACC (%) results on DRCT-2M.

Method	LCM-SD1.5	LCM-SDXL	LDM	SD-Turbo	SD1-Ctrl	SD1-DR	SD1.4	SD1.5	SD2	SD2-Ctrl	SD2-DR	SDXL	SDXL-Ctrl	SDXL-DR	SDXL-RefinerTurbo	SDXL-	Avg.
NPR	49.58	49.46	49.78	49.47	49.48	97.31	49.50	49.85	59.46	47.92	96.32	52.46	50.36	95.15	48.46	50.47	59.06
UnivFD	54.50	65.84	85.42	56.46	67.95	64.61	56.84	56.43	58.16	65.41	56.14	63.22	75.84	53.89	55.05	56.46	62.01
FatFormer	48.63	50.10	53.33	48.59	49.91	64.42	48.54	48.55	48.57	50.11	59.54	48.55	61.10	54.18	48.61	48.54	51.95
SAFE	50.13	49.95	50.28	49.96	50.05	50.00	49.94	98.18	49.98	50.00	98.47	49.92	54.74	97.25	50.14	49.95	59.31
C2P-CLIP	50.36	53.12	59.58	50.55	51.06	50.87	55.60	76.29	50.52	53.20	59.62	52.25	68.58	56.03	50.84	50.25	55.55
AIDE	63.03	75.19	75.16	61.60	51.53	70.50	52.61	56.66	72.68	60.05	64.87	54.44	54.52	88.24	61.81	65.35	64.27
DRCT	91.12	94.96	96.69	93.38	96.21	96.28	95.55	94.00	94.84	92.63	69.52	96.19	91.90	57.42	93.41	92.89	90.44
AlignedForensics	99.76	94.80	99.94	99.92	99.97	99.97	99.97	99.97	99.86	99.28	99.93	87.62	86.09	97.12	69.36	91.27	95.30
DDA	89.20	76.88	93.52	90.99	92.50	72.68	92.87	92.29	84.43	94.71	76.91	61.94	96.73	60.26	98.48	88.45	85.18
Ours	97.07	95.51	98.90	95.39	98.57	98.54	98.03	97.94	92.37	98.66	98.70	80.69	98.84	96.54	98.41	95.27	96.21

DRCT-2M. As shown in Table 6, our method achieves the highest average accuracy of 96.21%, outperforming the strongest baseline, AlignedForensics, by 0.91 percentage points, and DDA by 11.03 percentage points. It obtains the best results on LCM-SDXL, SDXL-Ctrl, and SDXL-Turbo, while achieving at least 92.37% accuracy on 15 of the 16 subsets. Although the performance on SDXL is relatively lower, the consistently strong results across the remaining variants demonstrate robust generalization across diverse diffusion generators and generation settings.

Table 7: Detailed ACC (%) results on ForenSynths. WFIR denotes WhichFaceIsReal.

Method	Big-GAN	CRN	Cycle-GAN	Deepfake	Gau-GAN	IMLE	Pro-GAN	SAN	Seeing-Dark	Star-GAN	Style-GAN	Style-GAN2	WFIR	Avg.
NPR	54.95	0.40	62.38	29.39	35.96	4.13	51.03	47.95	58.61	49.90	50.31	50.14	49.80	41.92
UnivFD	87.50	55.67	96.74	69.44	98.79	68.11	99.38	58.22	62.22	95.12	79.99	69.41	69.20	77.68
FatFormer	96.90	69.32	95.28	86.05	95.17	69.45	97.21	62.10	80.83	89.75	87.51	81.63	87.50	84.52
SAFE	48.95	0.11	52.99	49.18	42.60	0.10	49.96	50.46	35.83	49.95	49.99	49.99	49.10	40.71
C2P-CLIP	92.00	92.77	99.32	91.88	98.20	97.69	97.06	74.20	87.50	99.70	76.96	58.51	65.40	87.01
AIDE	43.80	40.62	46.97	40.54	40.68	23.05	41.91	49.54	36.11	38.92	40.85	37.17	40.70	40.07
DRCT	81.35	78.35	90.73	51.32	93.84	82.62	71.05	84.93	72.22	53.00	62.74	63.80	73.85	73.83
AlignedForensics	50.60	49.97	49.17	54.89	50.21	49.97	50.54	59.13	54.17	51.83	51.89	50.63	50.00	51.77
DDA	86.02	57.10	71.54	59.15	91.24	57.10	95.86	84.49	90.28	58.05	91.86	90.73	79.35	77.91
Ours	94.70	82.73	88.35	76.36	95.64	76.72	92.50	85.20	79.22	75.03	92.80	95.10	94.55	86.84

ForenSynths. As shown in Table 7, our method achieves an average accuracy of 86.84%, performing comparably to the strongest baseline, C2P-CLIP (87.01%), with a gap of only 0.17 percentage points. It outperforms DDA by 8.93 percentage points and obtains the best results on SAN, StyleGAN, StyleGAN2, and WFIR. Moreover, our method maintains at least 75.03% accuracy across all subsets, demonstrating balanced generalization across heterogeneous synthesis sources.

Table 8: Detailed ACC (%) results on EvalGEN, WildRF, and Chameleon.

Method	EvalGEN						WildRF				Chameleon
	Flux	GoT	Infinity	NOVA	Omni-Gen	Avg.	Facebook	Reddit	Twitter	Avg.	
NPR	30.80	30.80	30.91	30.89	30.87	30.85	50.00	78.07	37.29	55.12	59.44
UnivFD	33.60	37.90	41.39	58.25	36.21	41.47	49.06	60.07	45.84	51.66	57.39
FatFormer	31.32	41.34	43.33	93.48	35.20	48.93	54.38	69.67	42.76	55.60	58.03
SAFE	31.77	31.41	32.38	31.49	32.16	31.84	50.94	74.07	37.53	54.18	59.19
C2P-CLIP	32.32	35.70	38.94	63.19	32.92	40.61	51.56	71.53	40.62	54.57	57.59
AIDE	65.59	69.01	47.68	65.90	50.65	59.77	50.94	59.20	46.08	52.07	59.97
DRCT	79.90	84.25	83.62	85.58	80.52	82.77	45.94	53.00	65.80	54.91	51.63
AlignedForensics	35.64	62.07	77.47	81.55	79.81	67.31	95.63	73.71	77.91	82.41	70.20
DDA	43.90	79.13	62.35	96.84	86.46	73.74	74.38	82.60	77.45	78.14	73.03
Ours	74.18	96.34	84.78	96.87	96.61	89.76	88.75	88.73	90.63	89.37	74.24

EvalGEN, WildRF, and Chameleon. As shown in Table 8, our method achieves the best overall performance on all three benchmarks, reaching 89.76% on EvalGEN, 89.37% on WildRF, and 74.24% on Chameleon. These results exceed the corresponding strongest baselines by 6.99, 6.96, and 1.21 percentage points, respectively. Our method also ranks first on four of the five EvalGEN generators and two of the three WildRF sources, demonstrating strong generalization to recent generators and real-world social-media images.

Table 9: Detailed ACC (%) results on Synthbuster.

Method	DALL-E2	DALL-E3	Firefly	GLIDE	Midjourney	SD1.3	SD1.4	SD2	SDXL	Avg.
NPR	52.58	3.75	3.05	49.02	49.97	51.13	50.63	39.72	52.68	39.17
UnivFD	70.99	34.82	77.39	40.72	39.97	57.88	57.33	63.18	55.43	55.30
FatFormer	53.53	35.32	59.83	67.28	43.87	52.43	53.18	51.03	66.08	53.62
SAFE	64.28	16.21	16.56	58.48	63.03	65.73	65.43	59.33	65.78	52.76
C2P-CLIP	54.88	43.62	81.49	81.64	52.73	79.69	79.94	65.63	72.89	68.06
AIDE	34.47	41.77	24.16	56.53	47.17	71.49	71.29	49.92	61.28	50.90
DRCT	57.73	67.13	64.63	63.18	54.23	67.13	67.18	63.78	51.83	61.87
AlignedForensics	49.92	49.52	49.82	58.03	99.50	99.50	99.55	99.30	98.90	78.23
DDA	64.70	88.30	64.20	89.65	86.80	87.60	88.30	91.00	96.80	84.15
Ours	88.60	91.95	75.30	91.20	92.25	95.40	95.40	93.65	95.00	90.97

Synthbuster. As shown in Table 9, our method achieves the highest average accuracy of 90.97%, outperforming the strongest baseline, DDA, by 6.82 percentage points. It obtains the best results on DALL-E2, DALL-E3, and GLIDE, while maintaining at least 75.30% accuracy across all nine generators. Compared with baselines that perform strongly only on particular generator families, our method provides more consistent performance, demonstrating robust cross-generator generalization.

C.2 Detailed AUC Results

We further report detailed AUC results on the evaluated benchmarks. For benchmarks containing multiple generators or data sources, we present the result on each subset together with the macro average. These threshold-independent results complement the detailed ACC comparisons reported above.

Overall AUC comparison. As shown in Table 10, our method achieves the highest overall mean AUC of 95.54%, outperforming the strongest baseline, DDA, by 2.76 percentage points. It ranks first on five of the nine benchmarks, including AIGCDB, WildRF, Synthbuster, SynthWildX, and GenImage, and ranks second on DRCT-2M, ForenSynths, and EvalGEN. Moreover, our method achieves an AUC above 94.5% on eight benchmarks, with Chameleon being the only exception. These results demonstrate consistently strong threshold-independent generalization across diverse generators, synthesis paradigms, and real-world data distributions.

Table 10: Average AUC (%) across nine benchmarks. Mean is the unweighted average of the benchmark-level results. AIGCDB denotes AIGCDetectionBenchmark.

Method	AIGCDB	DRCT-2M	WildRF	ForenSynths	EvalGEN	Synthbuster	SynthWildX	Chameleon	GenImage	Mean
NPR	40.03	49.45	60.84	30.23	65.71	37.76	45.46	35.27	51.03	46.20
UnivFD	87.87	86.74	62.40	91.95	80.41	53.86	46.53	55.37	83.81	72.11
FatFormer	76.42	35.11	72.10	95.57	62.67	51.58	60.09	58.60	63.94	64.01
SAFE	46.89	55.38	62.10	39.55	53.98	59.59	49.05	57.13	53.69	53.04
C2P-CLIP	84.83	85.44	60.57	97.04	75.79	62.43	56.43	39.98	81.17	71.52
AIDE	63.86	82.99	57.25	42.73	87.57	49.92	57.69	64.97	78.91	65.10
DRCT	91.52	96.50	53.17	89.48	95.78	75.34	58.35	71.88	94.84	80.76
AlignedForensics	77.10	99.91	87.29	64.13	99.55	74.95	87.06	82.93	89.46	84.71
DDA	96.27	96.88	88.48	94.95	96.72	92.66	89.61	82.98	96.47	92.78
Ours	98.02	99.43	95.60	96.59	97.70	97.35	94.54	82.52	98.11	95.54

Table 11: Detailed AUC (%) results on AIGCDetectionBenchmark. WFIR denotes WhichFaceIsReal.

Method	ADM	BigGAN	CycleGAN	DALL-E2	GauGAN	GLIDE	Midjourney	ProGAN	SDXL	SD1.4	SD1.5	StarGAN	StyleGAN	StyleGAN2	VQDM	WFIR	Wukong	Avg.
NPR	43.70	37.26	67.11	34.69	17.48	43.65	39.80	46.97	39.71	47.79	47.45	10.68	33.57	31.56	48.58	40.25	50.29	40.03
UnivFD	83.55	97.03	99.54	59.52	99.90	84.35	74.21	99.99	85.09	78.03	77.57	98.90	95.30	96.61	92.72	87.86	83.63	87.87
FatFormer	66.57	99.53	99.96	16.96	99.92	68.70	45.95	99.99	71.00	47.59	47.70	100.00	97.39	95.53	82.01	97.93	62.38	76.42
SAFE	53.60	40.70	62.71	53.00	22.73	66.19	42.92	38.40	43.27	42.55	42.46	65.02	44.93	40.50	53.15	38.27	46.82	46.89
C2P-CLIP	74.01	98.53	99.93	39.30	99.85	88.80	71.12	99.97	74.07	73.86	71.87	100.00	96.47	92.92	88.10	93.62	79.77	84.83
AIDE	59.17	77.31	76.79	85.33	71.51	47.06	65.42	30.85	42.80	76.74	92.00	92.14	46.92	48.76	38.99	44.16	89.64	63.86
DRCT	88.95	91.17	98.10	96.89	98.42	95.49	93.05	90.53	97.05	96.85	96.74	92.46	71.20	72.88	95.06	84.36	96.70	91.52
AlignedForensics	79.56	40.07	37.48	96.73	48.48	83.38	99.79	83.74	100.00	99.93	99.99	83.79	68.69	70.99	94.11	23.95	99.99	77.10
DDA	97.34	97.30	92.43	99.52	98.26	97.37	93.74	99.62	99.81	99.82	99.80	70.47	99.82	99.63	93.47	98.48	99.77	96.27
Ours	96.93	98.58	95.92	99.57	98.25	97.62	94.90	99.86	99.84	99.82	99.77	93.18	99.64	99.15	95.28	98.29	99.82	98.02

AIGCDetectionBenchmark. As shown in Table 11, our method achieves the highest average AUC of 98.02%, outperforming the strongest baseline, DDA, by 1.75 percentage points. It obtains the best results on DALL-E2, GLIDE, and VQDM, while maintaining an AUC of at least 93.18% across all 17 subsets. These consistently high results demonstrate strong threshold-independent discriminability across diverse generators.

Table 12: Detailed AUC (%) results on DRCT-2M.

Method	LCM-SD1.5	LCM-SDXL	LDM	SD-Turbo	SD1-Ctrl	SD1-DR	SD1.4	SD1.5	SD2	SD2-Ctrl	SD2-DR	SDXL	SDXL-Ctrl	SDXL-DR	SDXL-RefinerTurbo	SDXL-Turbo	Avg.
NPR	42.30	29.59	44.48	27.88	37.76	99.71	42.88	42.40	41.89	32.28	99.35	23.48	46.43	99.35	53.47	27.88	49.45
UnivFD	81.03	91.63	98.26	83.83	92.19	89.14	85.42	84.96	87.50	90.96	80.11	89.69	93.33	76.24	79.76	83.83	86.74
FatFormer	18.05	29.74	70.43	12.86	26.03	74.86	16.26	15.58	14.88	32.52	68.26	15.26	65.21	59.92	31.35	10.49	35.11
SAFE	48.20	48.94	50.48	36.14	43.57	43.11	32.68	99.88	38.62	41.27	99.95	48.84	56.81	99.84	46.04	51.76	55.38
C2P-CLIP	78.65	89.19	96.24	79.63	83.14	82.62	91.53	98.30	85.75	87.66	79.51	89.47	93.27	64.38	90.04	77.61	85.44
AIDE	84.03	87.65	87.98	82.58	77.38	89.50	74.93	78.29	86.52	84.62	83.76	77.74	79.83	92.10	78.91	82.08	82.99
DRCT	97.60	98.90	99.71	98.35	99.46	99.45	99.19	98.62	98.94	98.08	86.42	99.39	97.86	75.57	98.33	98.17	96.50
AlignedForensics	99.98	99.98	100.00	99.97	99.98	99.97	99.93	100.00	100.00	99.92	99.98	99.97	99.64	99.80	99.42	99.97	99.91
DDA	98.91	95.88	99.44	99.00	99.09	94.42	99.30	99.16	97.85	99.49	95.72	86.06	99.67	87.54	99.89	98.64	96.88
Ours	99.64	99.41	100.00	99.12	99.95	99.93	99.83	99.85	98.81	99.95	99.96	95.80	99.99	99.58	99.78	99.30	99.43

DRCT-2M. As shown in Table 12, our method achieves an average AUC of 99.43%, ranking second among all compared methods and trailing AlignedForensics by only 0.48 percentage points. It outperforms DDA by 2.55 percentage points, obtains the best results on SD2-Ctrl and SDXL-Ctrl, and ties for the best result on LDM. Moreover, our method maintains an AUC of at least 95.80% across all 16 subsets, demonstrating consistently strong threshold-independent discriminability across diverse diffusion generators and generation settings.

Table 13: Detailed AUC (%) results on ForenSynths. WFIR denotes WhichFaceIsReal.

Method	Big-GAN	CRN	Cycle-GAN	Deepfake	Gau-GAN	IMLE	Pro-GAN	SAN	Seeing-Dark	Star-GAN	Style-GAN	Style-GAN2	WFIR	Avg.
NPR	37.26	0.80	67.11	16.68	17.48	13.65	46.97	35.23	41.48	10.68	33.57	31.56	40.55	30.23
UnivFD	97.03	96.74	99.54	81.27	99.90	98.38	99.99	79.16	64.71	98.90	95.30	96.61	87.86	91.95
FatFormer	99.53	92.33	99.96	95.80	99.92	99.09	99.99	67.92	97.07	100.00	97.39	95.53	97.93	95.57
SAFE	40.70	0.00	62.71	58.46	22.73	0.00	38.40	65.66	36.83	65.02	44.93	40.50	38.27	39.55
C2P-CLIP	98.53	98.01	99.93	97.70	99.85	99.60	99.97	93.16	91.76	100.00	96.47	92.92	93.62	97.04
AIDE	47.06	40.74	65.42	42.12	30.85	13.86	42.80	62.52	31.35	46.92	48.76	38.99	44.16	42.73
DRCT	91.17	89.92	98.10	89.34	98.42	98.17	90.53	93.03	93.63	92.46	71.20	72.88	84.36	89.48
AlignedForensics	40.07	62.33	37.48	90.99	48.48	68.73	83.74	97.50	56.93	83.79	68.69	70.99	23.95	64.13
DDA	96.00	99.96	92.34	82.10	98.06	99.93	99.24	93.50	97.50	83.26	97.31	96.95	98.16	94.95
Ours	99.17	99.26	96.97	94.36	99.51	99.34	99.55	90.94	93.74	85.50	99.09	98.97	99.32	96.59

ForenSynths. As shown in Table 13, our method achieves an average AUC of 96.59%, ranking second among all compared methods and trailing C2P-CLIP by only 0.45 percentage points. It outperforms DDA by 1.64 percentage points and achieves the best results on StyleGAN, StyleGAN2, and WFIR. Furthermore, our method maintains an AUC of at least 85.50% across all 13 subsets, demonstrating consistently strong discriminability across heterogeneous synthesis techniques.

Table 14: Detailed AUC (%) results on EvalGEN, WildRF, and Chameleon.

Method	EvalGEN						WildRF				Chameleon
	Flux	GoT	Infinity	NOVA	Omni-Gen	Avg.	Facebook	Reddit	Twitter	Avg.	
NPR	61.63	73.93	61.62	66.62	64.73	65.71	47.84	81.79	52.88	60.84	35.27
UnivFD	69.04	80.06	82.43	92.07	78.47	80.41	44.44	66.99	75.79	62.40	55.37
FatFormer	37.48	64.56	62.14	98.23	50.93	62.67	74.03	79.63	62.63	72.10	58.60
SAFE	53.08	56.57	49.52	53.76	56.96	53.98	51.04	78.53	56.74	62.10	57.13
C2P-CLIP	63.46	78.04	76.78	95.08	65.58	75.79	52.53	75.83	53.36	60.57	39.98
AIDE	89.25	90.71	81.71	89.60	86.60	87.57	54.91	63.32	53.51	57.25	64.97
DRCT	94.81	96.28	96.04	96.42	95.37	95.78	41.15	59.54	58.81	53.17	71.88
AlignedForensics	98.38	99.77	99.84	99.89	99.84	99.55	97.48	72.53	91.88	87.29	82.93
DDA	90.65	98.94	94.99	99.79	99.21	96.72	86.97	90.86	87.62	88.48	82.98
Ours	93.32	99.13	96.76	99.85	99.43	97.70	94.00	96.30	96.49	95.60	82.52

EvalGEN, WildRF, and Chameleon. As shown in Table 14, our method achieves average AUCs of 97.70%, 95.60%, and 82.52% on EvalGEN, WildRF, and Chameleon, respectively. It obtains the best performance on WildRF, outperforming the strongest baseline, DDA, by 7.12 percentage points, with the highest AUCs on both Reddit and Twitter. On EvalGEN, our method ranks second and exceeds DDA by 0.98 percentage points. It also remains competitive on Chameleon, with only a 0.46 percentage-point gap from the best result. These results demonstrate strong generalization to recent generators and real-world social-media images.

Table 15: Detailed AUC (%) results on Synthbuster.

Method	DALL-E2	DALL-E3	Firefly	GLIDE	Midjourney	SD1.3	SD1.4	SD2	SDXL	Avg.
NPR	47.87	1.29	0.27	48.53	49.31	49.47	50.44	36.22	56.43	37.76
UnivFD	78.59	9.02	87.77	34.40	28.83	60.70	60.63	67.70	57.11	53.86
FatFormer	57.13	18.63	58.58	71.83	37.44	48.55	49.99	51.29	70.74	51.58
SAFE	74.51	3.69	4.94	66.33	75.61	82.27	82.14	67.30	79.49	59.59
C2P-CLIP	57.31	0.17	87.70	82.71	33.61	83.94	83.78	60.43	72.23	62.43
AIDE	30.99	38.17	12.09	54.45	44.64	80.59	79.06	48.23	61.03	49.92
DRCT	60.11	88.91	78.10	74.10	57.92	91.97	91.76	80.93	54.26	75.34
AlignedForensics	50.42	14.62	45.18	64.74	100.00	99.96	100.00	99.83	99.84	74.95
DDA	77.39	96.62	77.47	96.10	96.22	96.26	96.45	97.68	99.75	92.66
Ours	95.88	97.51	88.61	97.59	98.19	99.95	99.96	98.74	99.75	97.35

Table 17: Average AP (%) across nine benchmarks. Mean is the unweighted average of the benchmark-level results. AIGCDB denotes AIGCDetectionBenchmark.

Method	AIGCDB	DRCT-2M	WildRF	ForenSynths	EvalGEN	Synthbuster	SynthWildX	Chameleon	GenImage	Mean
NPR	48.41	53.79	69.15	47.57	72.99	45.19	46.09	40.91	46.72	50.31
UnivFD	87.13	86.20	54.11	91.75	88.82	60.29	51.37	47.68	82.54	75.83
FatFormer	79.63	46.50	77.88	96.22	78.77	55.81	58.25	57.80	68.03	67.12
SAFE	49.87	57.65	70.66	45.43	74.06	57.35	48.51	50.64	56.13	54.82
C2P-CLIP	85.80	85.34	69.17	97.47	87.71	71.31	55.84	40.19	82.22	76.83
AIDE	59.36	74.70	59.10	44.74	88.80	52.20	53.19	51.75	72.46	62.69
DRCT	90.42	96.12	59.36	88.57	97.72	74.53	54.08	66.40	93.93	83.89
AlignedForensics	77.51	99.71	91.93	65.24	97.95	78.19	87.77	80.21	88.55	84.86
DDA	95.34	96.84	88.83	94.64	98.14	93.09	88.52	81.13	96.65	93.38
Ours	97.93	99.28	95.57	95.80	98.63	97.48	94.37	81.35	98.33	95.42

Synthbuster. As shown in Table 15, our method achieves the highest average AUC of 97.35%, outperforming the strongest baseline, DDA, by 4.69 percentage points. It obtains the best results on DALL-E2, DALL-E3, Firefly, and GLIDE, while remaining highly competitive on the other subsets. Notably, our method maintains an AUC of at least 88.61% across all nine generators, demonstrating robust threshold-independent discriminability against diverse and unseen image synthesis models.

Table 16: Generator-wise AUC (%) results on GenImage and SynthWildX.

Method	GenImage									SynthWildX			
	ADM	Big-GAN	GLIDE	Mid-journey	SD1.4	SD1.5	VQDM	Wukong	Avg.	DALL-E3	Firefly	Mid-journey	Avg.
NPR	50.77	62.21	47.51	41.40	48.61	48.09	57.06	52.59	51.03	37.21	52.40	46.78	45.46
UnivFD	83.55	96.43	84.35	74.21	78.03	77.57	92.72	83.63	83.81	34.32	73.11	32.17	46.53
FatFormer	66.57	90.61	68.70	45.95	47.59	47.70	82.01	62.38	63.94	48.60	73.78	57.89	60.09
SAFE	53.60	81.81	66.19	42.92	42.55	42.46	53.15	46.82	53.69	44.57	54.79	47.79	49.05
C2P-CLIP	74.01	99.64	88.80	71.12	74.07	73.86	88.10	79.77	81.17	45.82	73.80	49.66	56.43
AIDE	85.33	92.00	92.14	59.17	76.79	89.64	71.51	64.71	78.91	50.92	50.46	71.68	57.69
DRCT	88.95	95.57	95.49	93.05	97.05	96.85	95.06	96.70	94.84	63.41	59.89	51.76	58.35
AlignedForensics	79.56	58.96	83.38	99.79	100.00	99.93	94.11	99.99	89.46	89.68	72.56	98.95	87.06
DDA	95.86	97.68	97.95	95.97	98.18	98.13	89.53	98.46	96.47	93.99	83.73	91.12	89.61
Ours	97.54	98.98	98.03	94.03	99.91	99.87	96.63	99.85	98.11	96.37	91.59	95.67	94.54

GenImage and SynthWildX. As shown in Table 16, our method achieves the highest average AUCs on both GenImage and SynthWildX. On GenImage, it reaches an average AUC of 98.11%, outperforming the strongest baseline, DDA, by 1.64 percentage points, and obtains the best results on ADM, GLIDE, and VQDM. On SynthWildX, our method achieves an average AUC of 94.54%, exceeding DDA by 4.93 percentage points. It also delivers the best performance on DALL-E3 and Firefly while remaining competitive on Midjourney. These results demonstrate strong threshold-independent generalization across both established generators and recent real-world synthesis models.

C.3 Detailed AP Results

We further report detailed AP results on the evaluated benchmarks. For benchmarks containing multiple generators or data sources, we present the results on each subset together with the macro average.

Overall AP comparison. As shown in Table 17, our method achieves the highest overall mean AP of 95.42%, outperforming the strongest baseline, DDA, by 2.04 percentage points. It ranks first on seven of the nine benchmarks, including AIGCDB, WildRF, EvalGEN, Synthbuster, SynthWildX, Chameleon, and GenImage. It also achieves competitive performance on DRCT-2M and ForenSynths, where it remains among the top-performing methods. These results demonstrate consistently strong precision-recall performance across diverse generators, synthesis paradigms, and real-world data distributions.

Table 18: Detailed AP (%) results on AIGCDetectionBenchmark. WFIR denotes WhichFaceIsReal.

Method	ADM	Big-GAN	Cycle-GAN	DALL-E2	Gau-GAN	GLIDE	Mid-journey	Pro-GAN	SDXL	SD1.4	SD1.5	Star-GAN	Style-GAN	Style-GAN2	VQDM	WFIR	Wu-kong	Avg.
NPR	50.77	45.28	71.84	44.07	36.91	47.51	41.40	54.07	41.83	48.61	48.09	44.82	45.87	44.06	57.06	48.16	52.59	48.41
UnivFD	82.85	97.39	99.56	55.43	99.91	82.81	72.49	99.99	85.57	75.28	74.92	98.99	95.89	96.60	92.92	88.14	82.57	87.13
FatFormer	70.65	99.61	99.96	34.44	99.94	72.14	50.63	99.99	75.55	52.68	52.91	100.00	97.96	96.49	84.83	97.93	67.92	79.63
SAFE	54.23	44.17	66.55	53.17	34.97	68.54	45.26	43.46	45.38	47.45	47.37	57.62	48.10	42.91	55.94	40.33	52.27	49.87
C2P-CLIP	74.88	98.81	99.94	44.80	99.87	90.43	70.51	99.97	75.53	75.38	73.39	100.00	96.81	92.77	89.18	94.10	82.22	85.80
AIDE	56.49	68.75	70.17	76.56	62.84	45.08	56.05	37.60	42.81	66.03	86.58	86.74	43.57	44.51	40.20	42.90	82.29	59.36
DRCT	87.64	90.87	97.91	96.12	98.33	94.57	92.15	90.39	96.50	96.18	96.19	92.65	68.63	69.24	94.06	79.76	96.00	90.42
AlignedForensics	74.00	48.60	41.51	96.12	53.33	80.81	99.80	84.10	100.00	99.96	99.99	82.69	61.79	64.05	94.34	36.56	99.99	77.51
DDA	96.13	96.03	91.86	98.36	97.95	98.02	95.80	99.52	99.87	98.32	98.36	72.04	99.37	97.22	88.99	94.57	98.34	95.34
Ours	97.19	98.46	95.71	99.54	98.11	97.66	94.62	99.87	99.75	99.75	99.68	92.34	99.49	99.00	95.49	98.38	99.71	97.93

AIGCDetectionBenchmark. As shown in Table 18, our method achieves the highest average AP of 97.93%, outperforming the strongest baseline, DDA, by 2.59 percentage points. It obtains the best results on ADM, DALL-E2, StyleGAN, StyleGAN2, VQDM, and WFIR, while maintaining an AP of at least 92.34% across all 17 subsets. These results demonstrate consistently strong precision-recall performance across diverse image generators.

Table 19: Detailed AP (%) results on DRCT-2M.

Method	LCM-SD1.5	LCM-SDXL	LDM	SD-Turbo	SD1-Ctrl	SD1-DR	SD1.4	SD1.5	SD2	SD2-Ctrl	SD2-DR	SDXL	SDXL-Ctrl	SDXL-DR	SDXL-Refiner	SDXL-Turbo	Avg.
NPR	42.67	42.06	44.54	42.14	40.92	99.59	42.59	42.44	41.90	40.94	99.27	45.05	48.18	99.20	47.02	42.14	53.79
UnivFD	79.40	91.51	98.27	82.70	92.27	89.49	84.04	83.61	86.24	90.99	79.82	89.49	94.11	75.57	79.00	82.70	86.20
FatFormer	33.76	40.36	68.76	32.52	38.82	78.28	33.29	33.15	32.95	42.40	71.85	33.03	72.06	62.26	38.36	32.11	46.50
SAFE	50.74	49.07	52.92	40.17	46.75	46.26	38.73	99.91	42.83	43.75	99.96	47.09	64.27	99.87	50.34	49.77	57.65
C2P-CLIP	76.58	89.41	96.01	77.49	81.35	80.66	91.38	98.32	84.49	87.46	82.90	89.05	94.26	71.82	89.34	74.93	85.34
AIDE	74.14	81.93	82.03	74.10	67.48	80.30	66.11	69.45	79.06	76.77	74.78	68.36	69.71	84.12	71.20	75.62	74.70
DRCT	97.39	98.78	99.72	98.15	99.45	99.44	99.15	98.55	98.87	97.89	85.38	99.35	97.58	72.14	98.18	97.91	96.12
AlignedForensics	99.98	99.16	99.92	100.00	100.00	100.00	99.93	97.52	100.00	99.94	99.99	99.96	99.69	99.85	99.44	99.96	99.71
DDA	98.87	95.80	99.40	99.01	99.12	94.38	99.28	99.16	97.82	99.50	95.69	86.23	99.66	86.96	99.87	98.64	96.84
Ours	99.03	99.13	99.99	99.07	99.88	99.90	99.84	99.83	98.74	99.93	99.97	95.51	99.98	99.43	99.43	98.86	99.28

DRCT-2M. As shown in Table 19, our method achieves an average AP of 99.28%, ranking second among all compared methods and trailing AlignedForensics by only 0.43 percentage points. It outperforms DDA by 2.44 percentage points and obtains the best results on LDM and SDXL-Ctrl. Moreover, our method maintains an AP of at least 95.51% across all 16 subsets, demonstrating consistently strong precision-recall performance across diverse diffusion generators and generation settings.

Table 20: Detailed AP (%) results on ForenSynths. WFIR denotes WhichFaceIsReal.

Method	Big-GAN	CRN	Cycle-GAN	Deepfake	Gau-GAN	IMLE	Pro-GAN	SAN	Seeing-Dark	Star-GAN	Style-GAN	Style-GAN2	WFIR	Avg.
NPR	45.28	38.70	71.84	38.82	36.91	45.93	54.07	42.59	60.86	44.82	45.87	44.50	48.16	47.57
UnivFD	97.39	96.45	99.56	82.81	99.91	98.48	99.99	77.47	61.12	98.99	95.89	96.60	88.14	91.75
FatFormer	99.61	91.37	99.96	96.85	99.94	99.05	99.99	74.21	97.51	100.00	97.96	96.49	97.93	96.22
SAFE	44.17	30.69	66.55	50.00	34.97	30.69	43.46	61.50	39.62	57.62	48.10	42.91	40.33	45.43
C2P-CLIP	98.81	98.24	99.94	98.18	99.87	99.68	99.97	94.52	94.28	100.00	96.81	92.77	94.10	97.47
AIDE	45.08	55.14	56.05	43.24	37.60	33.42	42.81	58.77	38.27	43.57	44.51	40.20	42.90	44.74
DRCT	90.87	90.37	97.91	89.46	98.33	98.12	90.39	92.23	93.43	92.65	68.63	69.24	79.76	88.57
AlignedForensics	48.60	65.38	41.51	91.04	53.33	59.99	84.10	97.59	61.48	82.69	61.79	64.05	36.56	65.24
DDA	96.42	92.97	92.11	83.46	98.27	99.94	99.29	95.07	97.81	82.09	97.60	97.25	98.00	94.64
Ours	98.83	99.02	95.22	93.49	99.04	99.21	98.45	92.67	93.52	80.52	98.29	97.92	99.21	95.80

ForenSynths. As shown in Table 20, our method achieves an average AP of 95.80%, ranking third among all compared methods and outperforming DDA by 1.16 percentage points. It obtains the best results on CRN, StyleGAN, StyleGAN2, and WFIR. Moreover, our method achieves an AP of at least 92.67% on 12 of the 13 subsets, demonstrating strong precision-recall performance across heterogeneous synthesis techniques.

Table 21: Detailed AP (%) results on EvalGEN, WildRF, and Chameleon.

Method	EvalGEN						WildRF				Chameleon
	Flux	GoT	Infinity	NOVA	Omni-Gen	Avg.	Facebook	Reddit	Twitter	Avg.	
NPR	70.65	78.10	71.11	73.55	71.54	72.99	50.89	85.72	70.85	69.15	40.91
UnivFD	81.33	88.72	90.48	96.15	87.42	88.82	39.54	65.12	57.68	54.11	47.68
FatFormer	60.79	81.28	80.68	99.26	71.84	78.77	75.64	80.67	77.32	77.88	57.80
SAFE	72.63	75.64	72.06	73.41	76.54	74.06	55.11	83.88	72.99	70.66	50.64
C2P-CLIP	80.77	89.07	88.70	97.88	82.13	87.71	54.98	80.40	72.13	69.17	40.19
AIDE	90.72	91.24	85.25	89.84	86.94	88.80	51.16	60.55	65.59	59.10	51.75
DRCT	97.24	97.90	97.94	98.11	97.38	97.72	45.48	58.84	73.77	59.36	66.40
AlignedForensics	92.06	99.88	97.92	99.95	99.92	97.95	98.21	81.32	96.26	91.93	80.21
DDA	94.51	99.35	97.37	99.90	99.56	98.14	86.85	91.66	87.98	88.83	81.13
Ours	95.78	99.42	98.35	99.93	99.67	98.63	94.16	96.03	96.53	95.57	81.35

EvalGEN, WildRF, and Chameleon. As shown in Table 21, our method achieves the highest AP on all three benchmarks, with average APs of 98.63%, 95.57%, and 81.35% on EvalGEN, WildRF, and Chameleon, respectively. It outperforms the strongest baseline on the three benchmarks by 0.49, 3.64, and 0.22 percentage points, respectively. At the subset level, our method obtains the best results on Infinity, Reddit, and Twitter, demonstrating strong precision-recall performance on recent generators and real-world social-media images.

Table 22: Detailed AP (%) results on Synthbuster.

Method	DALL-E2	DALL-E3	Firefly	GLIDE	Midjourney	SD1.3	SD1.4	SD2	SDXL	Avg.
NPR	48.70	30.89	31.95	49.25	49.63	49.68	50.24	42.96	53.46	45.19
UnivFD	79.55	31.94	89.10	40.39	37.79	65.04	63.90	72.65	62.23	60.29
FatFormer	55.31	33.90	61.33	75.06	41.51	52.98	53.45	53.84	74.89	55.81
SAFE	64.07	31.12	31.28	58.52	66.75	69.07	68.99	60.28	66.08	57.35
C2P-CLIP	59.74	30.73	90.65	87.84	46.90	87.56	87.51	70.95	79.92	71.31
AIDE	37.60	44.34	32.58	51.72	46.95	77.00	74.96	48.09	56.55	52.20
DRCT	56.87	88.08	76.04	70.26	57.18	92.27	91.89	82.71	55.49	74.53
AlignedForensics	51.45	33.05	47.37	72.11	100.00	99.97	100.00	99.90	99.87	78.19
DDA	79.18	96.33	78.72	96.73	96.25	96.39	96.59	97.84	99.76	93.09
Ours	96.49	97.50	88.88	97.66	98.22	99.95	99.96	98.87	99.77	97.48

Synthbuster. As shown in Table 22, our method achieves the highest average AP of 97.48%, outperforming the strongest baseline, DDA, by 4.39 percentage points. It obtains the best results on DALL-E2, DALL-E3, and GLIDE, while remaining highly competitive on the other generators. Moreover, our method maintains an AP of at least 88.88% across all nine subsets, demonstrating robust precision-recall performance against diverse and unseen image synthesis models.

Table 23: Generator-wise AP (%) results on GenImage and SynthWildX.

Method	GenImage									SynthWildX			
	ADM	BigGAN	GLIDE	Midjourney	SD1.4	SD1.5	VQDM	Wukong	Avg.	DALL-E3	Firefly	Midjourney	Avg.
NPR	43.70	52.54	43.65	39.80	47.79	47.45	48.58	50.29	46.72	42.03	50.21	46.03	46.09
UnivFD	82.85	96.50	82.81	72.49	75.28	74.92	92.92	82.57	82.54	39.70	74.55	39.86	51.37
FatFormer	70.65	92.48	72.14	50.63	52.68	52.91	84.83	67.92	68.03	45.49	74.39	54.87	58.25
SAFE	54.23	77.94	68.54	45.26	47.45	47.37	55.94	52.27	56.13	44.13	54.39	47.00	48.51
C2P-CLIP	74.88	99.64	90.43	70.51	75.53	75.38	89.18	82.22	82.22	44.73	75.55	47.24	55.84
AIDE	76.56	86.58	86.74	56.49	70.17	82.29	62.84	57.99	72.46	47.92	47.16	64.48	53.19
DRCT	87.64	94.31	94.57	92.15	96.50	96.18	94.06	96.00	93.93	56.23	55.25	50.77	54.08
AlignedForensics	74.00	59.53	80.81	99.80	100.00	99.96	94.34	99.99	88.55	92.60	71.67	99.03	87.77
DDA	96.22	97.80	98.14	95.75	98.20	98.12	90.54	98.42	96.65	91.65	84.00	89.92	88.52
Ours	97.99	99.11	98.77	94.64	99.91	99.90	96.40	99.90	98.33	95.06	92.14	95.92	94.37

GenImage and SynthWildX. As shown in Table 23, our method achieves the highest average AP on both GenImage and SynthWildX. On GenImage, it obtains an average AP of 98.33%, outperforming the strongest baseline, DDA, by 1.68 percentage points, and achieves the best results on ADM, GLIDE, and VQDM. On SynthWildX, our method reaches an average AP of 94.37%, exceeding DDA by 5.85 percentage points, with the best results on DALL-E3 and Firefly. These results demonstrate strong precision-recall performance across both established generators and recent real-world synthesis models.