# MODULO FOLDING: RETHINKING AI-GENERATED IMAGE DETECTION BEYOND LOW-BIT PLANES

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Reliable detection of AI-generated images is increasingly important for media forensics and trustworthy visual content. Recent studies often use image-based reconstruction errors as an important feature for determining whether an image is AI-generated. However, these approaches typically incur high computational costs and also fail to capture intrinsic noisy features present in the raw images. The recently proposed LOTA method assumes that low-bit planes provide the key discriminative cue and uses bit-plane guided noise generation for detection, which achieves good performance on AI-generated image detection. However, the underlying mechanism of LOTA remains unclear. In this paper, we reveal that LOTA's bit-plane guided noise generation is mathematically equivalent to taking the image modulo eight. This equivalence shows that the essential ingredient is not the bit-plane representation itself but periodic modulo folding. Indeed, we observe that a broad range of modulo periods achieve comparable performance. We provide a theoretical explanation for this phenomenon: modulo folding compresses the large-amplitude semantic component into a small remainder while preserving small-amplitude artifacts, thereby substantially increasing the relative variance and information of artifacts. Based on this finding, we propose Multi-Period Modulo Folding Detector (MP-MFD), which generalizes the single-period modulo eight design of LOTA to multiple periods and injects the resulting folded images into different stages of a backbone network to capture complementary multi-level artifact cues. MP-MFD requires no reconstruction or iterative sampling and treats LOTA as a special case. Extensive experiments show that MP-MFD achieves SOTA results. These results validate the effectiveness of modulo folding.

## 1 INTRODUCTION

The rapid development of generative models, including GANs and diffusion models, has made it increasingly difficult to distinguish AI-generated images from real ones. This poses serious risks in misinformation, privacy, and digital forensics. As a result, AI-generated image detection has become an active research area. Early deep learning-based deepfake detectors were mainly designed for GAN-generated images. Nevertheless, recent studies Corvi et al. (2023); Ricker et al. (2022) reveal that the rise of diffusion models causes a notable performance drop when GAN-based detectors are directly applied to diffusion-generated images. To detect diffusion-generated images, numerous approaches based on reconstruction errors have been developed, including DIRE Wang et al. (2023), AEROBLADE Ricker et al. (2024), LaRE$^2$ Luo et al. (2024), and FIRE Chu et al. (2025). DIRE Wang et al. (2023) is the first to calculate the discrepancy between the original image and its diffusion-reconstructed counterpart, and treats images with smaller errors as AI-generated. LaRE$^2$ Luo et al. (2024) accelerates diffusion reconstruction by one step of the reverse diffusion process and further refines image features using reconstruction errors. FIRE Chu et al. (2025) incorporates frequency decomposition into reconstruction-based detection, thereby identifying image components that lead to larger reconstruction errors. GenSign Fei et al. (2026) utilizes diverse reconstruction residuals as supervision to learn generative signatures. However, these methods extract error signals via a diffusion reverse process, which is inefficient and also vulnerable to the reconstruction diffusion model.

A recent work, LOTA, introduces a different perspective. It decomposes each RGB channel into eight bit-planes, selects the lowest three bit-planes, and combines them into a noise image for detec-



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.

tion. LOTA achieves better results than these methods based on reconstruction errors, and attributes its success to the discovery that "low bit-planes represent noise patterns." However, LOTA does not explain why this operation works.

As shown in Figure 1, in this paper, we begin with a simple observation: combining the lowest three bit-planes is mathematically equivalent to taking the image modulo eight. In this sense, LOTA is a special case of modulo folding with a single fixed period of eight. More strikingly, we empirically find that replacing modulo eight with other modulo periods, such as six, seven, nine, ten, up to fourteen, also achieves good results on GenImage. This suggests that the effectiveness of LOTA does not stem from the specific choice of "bit-planes 0, 1, 2," but from a more general mechanism: periodic folding via modulo operation. This observation raises a fundamental question that the field has not yet answered. Why does modulo operation work for AI-generated image detection? What is the underlying mechanism that makes a simple per-pixel modulus so effective?

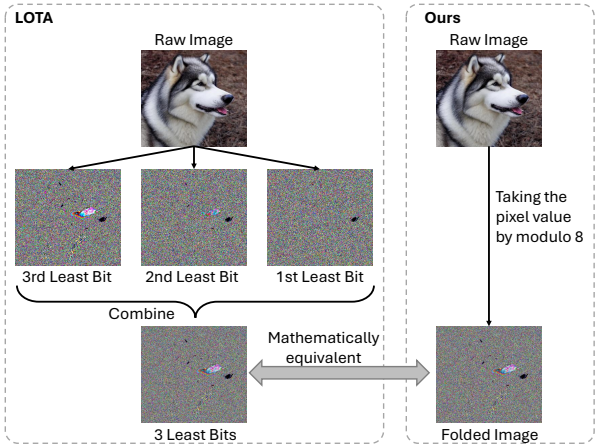


Figure 1: LOTA's core operation, i.e., combining the lowest three bit-planes, is mathematically equivalent to taking the image modulo eight.

We answer these questions by providing a new perspective on image formation and detection. We model an image as the superposition of a semantic component and an artifact component. The semantic component is large in amplitude, while the artifact component is small in amplitude. Directly separating them is ill-posed. However, taking the image modulo a period acts as a nonlinear periodic wrapping: the large semantic component is collapsed into a small interval, while the small artifact component is largely preserved as an additive perturbation. As a result, the relative contribution of artifacts is significantly amplified. This mechanism is fundamentally different from linear high-pass filtering, where semantic edges also lie in the high-frequency band. Modulo folding does not simply suppress low frequencies; it folds the entire semantic range into a compact periodic signal, making the artifact relatively more prominent.

Based on this insight, we propose Multi-Period Modulo Folding Detector (MP-MFD), which generalizes the single-period design to multiple periods and injects the resulting folded images into different stages of a backbone network. This multi-period, multi-stage design allows the detector to capture complementary artifact information at different levels.

Our contributions can be summarized as follows:

- We reveal that LOTA's bit-plane guided noise generation is mathematically equivalent to taking the image modulo eight, and that a broad range of modulo periods achieve comparable performance. This challenges the "bit-plane" narrative and identifies modulo folding as the essential ingredient.
- We provide a theoretical explanation for why modulo folding enhances artifact signals: it compresses the semantic dynamic range and variance, while preserving small artifact perturbations, thereby increasing the relative information of artifacts.
- We propose Multi-Period Modulo Folding Detector (MP-MFD), which fuses multiple periods folded images at different stages of a backbone network. This design captures multi-level artifact cues and generalizes LOTA beyond a single fixed modulo period.
- Extensive experiments show the effectiveness of MP-MFD, and that MP-MFD achieves SOTA results on GenImage, DRCT-2M, Co-Spy-Bench, and ForenSynths (in the Appendix) benchmarks.

# 2 Related Work

## 2.1 Spatial Domain-Based Methods

Methods in the spatial domain generally depend on local patterns or pixel-level statistics to separate real images from generated ones. Recently, Tan et al. Tan et al. (2024b) introduced neighboring pixel relationships (NPR), derived from local pixel differences, as a simple yet effective cue for detecting AI-generated images. Zhong et al. (2023); Chen et al. (2024c) exploit texture information by summing residuals along four directions, i.e., diagonal, counter-diagonal, horizontal, and vertical, for AI-generated image detection. Zheng et al. Zheng et al. (2024) adopted image patch shuffling to disturb semantic artifacts, thereby reducing the negative influence of image semantic information. MLEP Yuan et al. (2025) suggests employing image entropy, computed from shuffled small patches over multiple resampling scales, as a cue for detecting AI-generated images. CoD Jia et al. (2025) applies color quantization and restoration to derive color discrepancies for fake image detection.

## 2.2 Frequency Domain-Based Methods

To address the subtle nature of artifacts in AI-generated images, frequency-domain methods examine image frequency components, enabling more effective discrimination between real and fake images. Qian et al. Qian et al. (2020a) introduced a face forgery detection network that relies on frequency-aware decomposed image components and local frequency statistics. Luo et al. Luo et al. (2021) developed a feature representation based on multi-scale high-frequency noise and improved detection performance by integrating it with an attention module. Liu et al. Liu et al. (2022) adopted frequency-domain noise patterns as features for detecting AI-generated images. Tan et al. Tan et al. (2024a) proposed FreqNet, which exploits high-frequency representations across spatial and channel dimensions to improve detection generalizability.

## 2.3 Image Error-Based Methods

Recently, a growing number of studies have focused on errors between reconstructed images and original images. Wang et al. Wang et al. (2023) proposed Diffusion Reconstruction Error (DIRE), which distinguishes real images from DM-generated images by measuring reconstruction error. Ma et al. Ma et al. (2023) improved detection accuracy through multi-step error computation in their SeDID method. AEROBLADE Ricker et al. (2024) and RDD Choi et al. (2026) leverage autoencoder reconstruction error from latent DMs in a training-free manner. FakeInversion Cazenavette et al. (2024) detects images generated by unseen text-to-image DMs through text-conditioned inversion. Moreover, LaRE$^2$ Luo et al. (2024) leverages Latent Reconstruction Error (LaRE) together with an Error-Guided Feature Refinement module to extract more distinctive error features. FIRE Chu et al. (2025) incorporates frequency decomposition into the reconstruction process and identifies the image components responsible for higher reconstruction errors. Unlike the above approaches, we aim to uncover the forgery signal embedded in the image itself.

# 3 Method

## 3.1 Problem Formulation

Let $I(x)$ denote the intensity of an image at pixel location $x$. We model the image as the sum of two components:

$$I(x) = S(x) + A(x), \tag{1}$$

where $S(x)$ represents the semantic content, i.e., the meaningful visual structures that are shared across natural and generated images (objects, textures, lighting, and other scene-level information). It is typically large in amplitude. $A(x)$ represents the generation artifacts, i.e., the traces left by the generative process (upsampling residuals, denoising traces, frequency-domain periodicities, etc.). It is typically small in amplitude.

For real images, $A$ is approximately zero or contains only sensor noise; for AI-generated images, $A \neq 0$ and carries discriminative evidence. The central challenge is that $S$ and $A$ are highly entangled in the pixel domain. Directly separating $S(x)$ and $A(x)$ is ill-posed. Moreover, $S$ dominates

the image energy, i.e., $\|S\| \gg \|A\|$, so a naïve classifier trained on $I$ tends to focus on semantic content, which is largely shared between real and generated images.

Instead, in this work, we seek a simple nonlinear transform $R = \mathcal{T}(I)$ such that the relative contribution of $A$ in $R$ is significantly larger than that in $I$. We show that the modulo operation serves as an effective and theoretically grounded choice for $\mathcal{T}$. The key intuition is that modulo acts as a hard periodic folding: it aggressively suppresses the large-amplitude semantic component $S$, while preserving the small-amplitude artifact component $A$.

### 3.2 Modulo Folding as Periodic Wrapping

We define the modulo folding operator with a positive integer period $m$:

$$R(x) = I(x) \bmod m, \tag{2}$$

where $R(x) \in [0, m)$ is the folded image. To understand its effect, we decompose the semantic component $S(x)$ into its quotient and remainder with respect to $m$:

$$S(x) = mQ(x) + B(x), \tag{3}$$

where $Q(x) = \lfloor S(x)/m \rfloor$ is the quotient, indicating how many full periods $S(x)$ spans, and $B(x) = S(x) \bmod m \in [0, m)$ is the semantic remainder. Substituting this into the modulo operation yields

$$R(x) = (S(x) + A(x)) \bmod m = (B(x) + A(x)) \bmod m. \tag{4}$$

If the artifact magnitude satisfies $|A(x)| < m$ and the sum $B(x) + A(x)$ does not cross a multiple of $m$, then

$$R(x) = B(x) + A(x). \tag{5}$$

This simple derivation reveals the core mechanism: the large semantic component $S(x)$ is collapsed into its remainder $B(x)$, while the small artifact $A(x)$ is largely preserved as an additive perturbation. In other words, modulo folding compresses the semantic dynamic range, while retaining the artifact signal as soon as possible.

### 3.3 Why Modulo Folding Enhances Artifacts

We now provide a quantitative justification for why the modulo operation increases the relative information of artifacts. The analysis proceeds in three steps: artifact preservation probability, semantic dynamic range compression, and artifact variance ratio improvement.

#### 3.3.1 Artifact Preservation Probability

Whether an artifact is preserved without distortion depends on whether it pushes the remainder $B(x)$ across a period boundary. Assuming $B(x)$ is approximately uniformly distributed in $[0, m)$, the probability that the sum $B(x) + A(x)$ crosses a multiple of $m$ is

$$P_{\text{cross}} \approx \frac{|A(x)|}{m}. \tag{6}$$

Since artifacts are typically much smaller than the period, i.e., $|A(x)| \ll m$, we have

$$P_{\text{cross}} \ll 1. \tag{7}$$

Therefore, for the vast majority of pixels, the artifact component is not destroyed by modulo folding; it is simply added to the semantic remainder.

#### 3.3.2 Semantic Dynamic Range Compression

In the original image, the semantic component $S(x)$ can span the full intensity range, e.g., 0 to 255. After modulo folding, the semantic component becomes $B(x) = S(x) \bmod m$, whose dynamic range is restricted to

$$0 \le B(x) < m. \tag{8}$$

If $m \ll 255$, the dynamic range of the semantic content is drastically reduced. Meanwhile, the artifact amplitude $|A(x)|$ remains essentially unchanged. Consequently, the ratio of semantic dynamic range to artifact dynamic range changes from approximately

$$\frac{255}{|A|} \quad \text{to} \quad \frac{m}{|A|}. \tag{9}$$

Since $m \ll 255$, the semantic content no longer dominates the artifact in terms of amplitude.

### 3.3.3 Artifact Variance Ratio Improvement

We further analyze the effect in terms of variance. Let

- $\sigma_S^2 = \mathrm{Var}(S(x))$ be the variance of the original semantic component of the whole image;
- $\sigma_A^2 = \mathrm{Var}(A(x))$ be the variance of the artifact component of the whole image;
- $\sigma_B^2 = \mathrm{Var}(B(x))$ be the variance of the semantic remainder after modulo folding of the whole image.

In the original image, the artifact variance ratio is

$$\rho_{\text{orig}} = \frac{\sigma_A^2}{\sigma_S^2 + \sigma_A^2}. \tag{10}$$

After modulo folding, instead of assuming that the remainder $B(x)$ is uniformly distributed, we bound its variance by the maximum possible value. For $B(x) \in \{0, \ldots, m-1\}$, the variance is maximized when $B(x) = 0$ and $B(x) = m-1$ each occur with probability $1/2$. Thus

$$\sigma_{B,\max}^2 = \frac{1}{2}\left(0 - \frac{m-1}{2}\right)^2 + \frac{1}{2}\left((m-1) - \frac{m-1}{2}\right)^2 = \frac{(m-1)^2}{4}, \tag{11}$$

and this special case usually not occur in real world. Therefore, we have

$$\sigma_B^2 < \frac{(m-1)^2}{4}. \tag{12}$$

The artifact variance $\sigma_A^2$ is largely preserved because $|A(x)| \ll m$. Therefore, the new artifact variance ratio satisfies

$$\rho_{\text{mod}} = \frac{\sigma_A^2}{\sigma_B^2 + \sigma_A^2} > \frac{\sigma_A^2}{(m-1)^2/4 + \sigma_A^2}. \tag{13}$$

Comparing the two ratios, we obtain

$$\frac{\rho_{\text{mod}}}{\rho_{\text{orig}}} = \frac{\sigma_S^2 + \sigma_A^2}{\sigma_B^2 + \sigma_A^2} > \frac{\sigma_S^2 + \sigma_A^2}{(m-1)^2/4 + \sigma_A^2}. \tag{14}$$

Typically, $\sigma_S^2 \gg \sigma_A^2$ and $m$ is a tiny positive integer, we have

$$\rho_{\text{mod}} \gg \rho_{\text{orig}}. \tag{15}$$

This inequality formalizes the intuition: modulo folding compresses the semantic variance of the whole image from $\sigma_S^2$ to at most $(m-1)^2/4$, with equality in the worst case when $B(x) = 0$ and $B(x) = m-1$ each occur with probability $1/2$, while the artifact variance $\sigma_A^2$ is almost fully retained. Hence the relative variance contribution of artifacts is significantly amplified.

## 3.4 Multi-Period Modulo Folding Detector (MP-MFD)

Based on the above analysis, we propose Multi-Period Modulo Folding Detector (MP-MFD), which extends single-period modulo folding to multiple periods and fuses the resulting folded images at different stages of a backbone network. As shown in Figure 2, the pipeline consists of three steps.

**Multi-Period Folding**: Given an input image $I$, choose a set of periods $\{m_1, m_2, \ldots, m_K\}$, where $K$ is the number of periods. For each period $m_i$, compute the folded image

$$R_i = I \bmod m_i, \quad i = 1, 2, \ldots, K. \tag{16}$$

To feed $R_i$ into a standard CNN, we normalize it back to $[0, 255]$. We use the scaling strategies:

$$\tilde{R}_i = \left\lfloor \frac{255}{m_i - 1} \right\rfloor R_i, \quad i = 1, 2, \ldots, K. \tag{17}$$

where $\tilde{R}_i$ denotes the normalized folded image. The first folded image $\tilde{R}_1$ is used as the main network input, while $\tilde{R}_2, \ldots, \tilde{R}_K$ serve as auxiliary inputs.

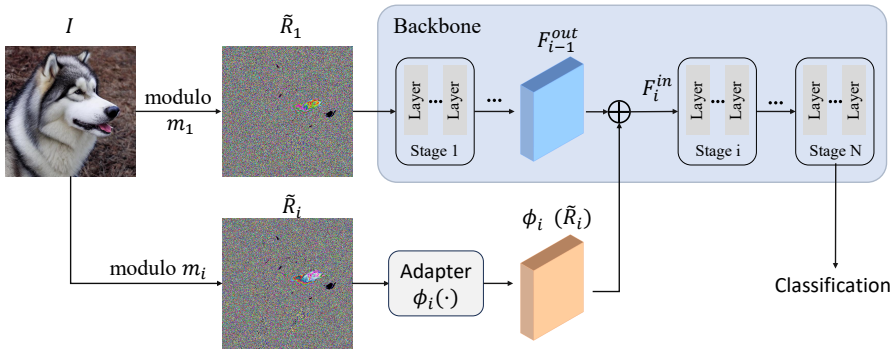


Figure 2: Framework overview of the proposed Multi-Period Modulo Folding Detector (MP-MFD).

| Method | Midjourney | SDv1.4 | SDv1.5 | ADM | GLIDE | Wukong | VQDM | BigGAN | Avg. |
|---|---|---|---|---|---|---|---|---|---|
| GramNet Liu et al. (2020) | 73.6 | 98.8 | 98.7 | 51.5 | 55.3 | 95.3 | 55.1 | 49.4 | 72.2 |
| UnivFD Ojha et al. (2023) | 91.4 | 96.4 | 96.1 | 58.0 | 73.4 | 94.5 | 67.8 | 57.7 | 79.4 |
| DIRE Wang et al. (2023) | 50.4 | 99.9 | 99.9 | 52.3 | 67.2 | 99.9 | 50.1 | 49.9 | 71.2 |
| LaRE$^2$ Luo et al. (2024) | 74.0 | 99.9 | 99.9 | 61.7 | 88.5 | 99.9 | 97.2 | 68.7 | 86.2 |
| DRCT Chen et al. (2024b) | 91.5 | 95.0 | 94.4 | 79.4 | 89.1 | 94.6 | 90.0 | 81.6 | 89.4 |
| ESSP Chen et al. (2024c) | 82.6 | 99.2 | 99.3 | 78.9 | 88.9 | 98.6 | 96.0 | 73.9 | 90.6 |
| Effort Yan et al. (2025) | 82.4 | 99.8 | 99.8 | 78.7 | 93.3 | 97.4 | 91.7 | 77.6 | 91.1 |
| DLFE Zhong et al. (2025) | 95.5 | 96.1 | 96.3 | 95.9 | 96.0 | 96.4 | 96.0 | 95.7 | 96.0 |
| GenSign Fei et al. (2026) | 99.3 | 99.8 | 99.7 | 84.3 | 90.4 | 99.8 | 99.5 | 99.8 | 96.6 |
| IAPL Li et al. (2026) | 95.1 | 99.9 | 99.8 | 85.4 | 95.9 | 99.8 | 98.8 | 98.7 | 96.7 |
| LOTA Wang et al. (2025) | 91.6 | 99.9 | 99.9 | 99.5 | 98.3 | 99.8 | 99.5 | 99.9 | 98.6 |
| MP-MFD | 98.5 | 99.9 | 99.9 | 99.0 | 99.0 | 99.1 | 99.0 | 98.9 | **99.2** |

Table 1: Results on GenImage Zhu et al. (2023a). Methods are trained on GenImage/SDv1.4 following Zhu et al. (2023a). The metric is Averaged Accuracy with 0.5 as the threshold.

**Multi-Stage Injection**: We adopt a backbone network with multiple stages, e.g., ResNet. The main folded image $\tilde{R}_1$ is fed into the network input and processed through the initial layers and Stage1. For each auxiliary folded image $\tilde{R}_i$ with $i \geq 2$, we introduce a lightweight adapter $\phi_i$ that projects $\tilde{R}_i$ to the same spatial resolution and channel dimension as the input feature of stage $i$. The adapters are implemented as several $3 \times 3$ Conv-BN-ReLU blocks, with appropriate stride when downsampling is needed. The projected feature is then fused with the stage input by element-wise addition:

$$F_i^{\text{in}} = F_{i-1}^{\text{out}} + \phi_i(\tilde{R}_i), \tag{18}$$

where $F_{i-1}^{\text{out}}$ is the output feature from the previous stage. $\phi_i(\tilde{R}_i)$ is the adapted feature from the $i$-th folded image. $F_i^{\text{in}}$ is the fused input to stage $i$. The fused feature $F_i^{\text{in}}$ is then processed by the next stage $i$.

**Classification**: After the final stage, global average pooling and a fully connected layer produce the binary classification output. The entire network is trained end-to-end with binary cross-entropy loss.

This design allows different modulo periods to contribute complementary artifact information at different depths of the network. The key advantage of MP-MFD is that it does not require any reconstruction, iterative sampling, or generative inversion. It operates directly on the original image and is therefore extremely efficient.

### 3.5 Relationship to Bit-Plane LOTA Methods

When $m = 2^k$, modulo folding is exactly equivalent to extracting the lowest $k$ bit-planes. In particular, $m = 8$ recovers the three lowest bit-planes used in LOTA. However, our framework reveals that the essential mechanism is not the bit-plane representation itself, but the periodic folding induced by the modulo operation. This explains why arbitrary moduli such as $5, 6, 7, 9, \ldots, 14$ work equally well: they all satisfy the conditions $m > \max|A|$ and $m \ll \sigma(S)$, and therefore achieve effective semantic suppression while preserving artifacts. Our method thus provides a unified theoretical foundation for bit-plane-based detection and extends it to a broader family of modulo operators.

| Methods | CNNSpot | GramNet | DeFake | UnivFD | DIRE | DRCT | DLFE | GenSign | IAPL | LOTA | **MP-MFD** |
|---|---|---|---|---|---|---|---|---|---|---|---|
| LDM | 99.9 | 99.4 | 92.1 | 98.3 | 98.2 | 96.7 | 97.1 | 97.4 | 97.7 | 98.9 | 99.5 |
| SDv1.4 | 99.9 | 99.0 | 99.5 | 96.2 | 99.9 | 96.3 | 97.1 | 99.0 | 99.3 | 99.9 | 99.9 |
| SDv1.5 | 99.9 | 98.8 | 99.5 | 96.3 | 99.9 | 96.3 | 97.1 | 97.0 | 98.0 | 99.5 | 99.7 |
| SDv2 | 97.5 | 95.3 | 89.7 | 93.8 | 68.2 | 94.9 | 97.1 | 96.6 | 96.4 | 98.2 | 99.7 |
| SDXL | 66.3 | 62.6 | 64.0 | 91.0 | 53.8 | 96.2 | 97.1 | 97.0 | 97.5 | 98.7 | 99.5 |
| SDXL-Refiner | 86.5 | 80.7 | 69.2 | 93.9 | 71.9 | 93.5 | 97.1 | 96.7 | 97.0 | 98.5 | 99.5 |
| SD-Turbo | 86.2 | 71.2 | 92.0 | 86.4 | 58.9 | 93.4 | 97.1 | 96.5 | 96.0 | 97.9 | 99.0 |
| SDXL-Turbo | 72.4 | 69.3 | 93.9 | 85.9 | 54.3 | 92.9 | 97.1 | 96.2 | 95.5 | 97.7 | 99.1 |
| LCM-SDv1.5 | 98.3 | 93.1 | 99.1 | 90.4 | 99.8 | 91.2 | 97.1 | 94.1 | 93.8 | 98.9 | 99.4 |
| LCM-SDXL | 61.7 | 57.0 | 70.9 | 89.0 | 59.7 | 95.0 | 97.1 | 96.2 | 95.2 | 98.3 | 99.3 |
| SDv1-Ctrl | 98.0 | 90.0 | 59.0 | 90.4 | 99.6 | 95.6 | 97.1 | 95.9 | 96.6 | 98.3 | 99.2 |
| SDv2-Ctrl | 85.9 | 75.6 | 62.3 | 81.1 | 64.2 | 92.7 | 96.8 | 96.2 | 95.7 | 98.6 | 99.3 |
| SDXL-Ctrl | 82.8 | 82.7 | 66.7 | 89.1 | 59.1 | 92.0 | 94.7 | 95.0 | 95.5 | 98.7 | 99.5 |
| SDv1-DR | 60.9 | 51.2 | 50.1 | 52.0 | 52.0 | 94.1 | 92.3 | 80.3 | 96.0 | 61.6 | 78.8 |
| SDv2-DR | 51.4 | 50.0 | 50.2 | 51.0 | 50.0 | 69.6 | 51.8 | 65.0 | 70.2 | 60.9 | 77.1 |
| SDXL-DR | 50.3 | 50.1 | 50.0 | 50.5 | 50.0 | 57.4 | 49.7 | 64.9 | 59.4 | 61.0 | 77.9 |
| Avg. | 81.1 | 76.6 | 75.5 | 83.5 | 71.2 | 90.5 | 90.9 | 91.5 | 92.5 | 91.6 | **95.4** |

Table 2: Results on DRCT-2M Chen et al. (2024b). Methods are trained on DRCT-2M/SDv1.4 following DRCT Chen et al. (2024b). The metric is Averaged Accuracy with 0.5 threshold. The compared methods are CNNSpot Wang et al. (2020), GramNet Liu et al. (2020), DeFake Sha et al. (2023), UnivFD Ojha et al. (2023), DIRE Wang et al. (2023), DRCT Chen et al. (2024b), DLFE Zhong et al. (2025), GenSign Fei et al. (2026), IAPL Li et al. (2026) and LOTA Wang et al. (2025).

| Methods | CNNSpot | LNP | UnivFD | DIRE | FreqNet | NPR | DRCT | CO-SPY | GenSign | IAPL | LOTA | **MP-MFD** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| LDM | 77.6 | 84.9 | 79.1 | 66.3 | 74.9 | 84.3 | 79.3 | 95.0 | 95.5 | 96.2 | 97.5 | 98.3 |
| SD-v1.4 | 90.0 | 95.9 | 80.9 | 83.5 | 69.2 | 90.9 | 81.2 | 92.0 | 93.8 | 94.3 | 99.3 | 99.5 |
| SD-v1.5 | 90.0 | 96.2 | 80.9 | 83.8 | 68.9 | 91.3 | 81.1 | 91.3 | 94.5 | 94.5 | 99.1 | 99.3 |
| SSD-1B | 66.6 | 79.1 | 76.5 | 56.6 | 49.2 | 47.9 | 75.8 | 83.2 | 92.1 | 92.0 | 96.3 | 98.4 |
| tiny-sd | 66.4 | 81.5 | 77.0 | 63.6 | 63.6 | 88.4 | 80.0 | 84.8 | 90.8 | 91.5 | 96.8 | 98.0 |
| SegMoE-SD | 74.4 | 86.6 | 83.1 | 66.0 | 64.4 | 93.8 | 75.1 | 89.5 | 92.9 | 93.8 | 97.2 | 98.2 |
| small-sd | 70.1 | 80.7 | 77.4 | 68.4 | 65.4 | 89.1 | 81.2 | 85.8 | 92.1 | 92.0 | 96.9 | 98.7 |
| SD-2-1 | 68.1 | 57.2 | 81.7 | 65.2 | 51.6 | 51.3 | 76.1 | 88.5 | 93.5 | 94.2 | 97.8 | 98.5 |
| SD-3-medium | 60.7 | 53.7 | 78.4 | 56.8 | 49.6 | 50.0 | 75.0 | 82.9 | 91.6 | 91.5 | 97.6 | 98.4 |
| SDXL-turbo | 88.7 | 83.5 | 84.3 | 73.0 | 66.6 | 83.6 | 80.4 | 95.4 | 94.1 | 95.0 | 97.9 | 98.8 |
| SD-2 | 65.7 | 54.6 | 73.8 | 60.1 | 50.6 | 51.2 | 75.1 | 83.7 | 90.3 | 91.7 | 97.8 | 98.8 |
| SDXL | 61.8 | 80.3 | 63.6 | 51.9 | 48.8 | 47.7 | 75.2 | 74.1 | 87.6 | 85.0 | 97.9 | 98.7 |
| PG-v2.5-1024 | 53.6 | 79.1 | 78.2 | 52.3 | 48.7 | 47.7 | 71.3 | 88.7 | 91.7 | 92.5 | 98.1 | 98.8 |
| PG-v2-1024 | 63.5 | 52.5 | 78.5 | 56.6 | 48.8 | 48.4 | 66.6 | 89.1 | 91.3 | 91.4 | 98.0 | 98.6 |
| PG-v2-512 | 57.9 | 49.4 | 58.9 | 53.6 | 49.1 | 49.1 | 77.6 | 64.9 | 88.5 | 88.0 | 97.5 | 98.6 |
| PG-v2-256 | 63.3 | 54.8 | 63.0 | 60.3 | 49.4 | 49.7 | 73.5 | 72.9 | 85.0 | 83.9 | 97.3 | 98.1 |
| PAXL-2-1024 | 56.3 | 53.4 | 80.1 | 54.8 | 49.4 | 51.2 | 71.8 | 93.9 | 89.8 | 90.5 | 96.8 | 98.6 |
| PAXL-2-512 | 65.4 | 74.2 | 80.3 | 62.2 | 57.2 | 81.3 | 75.5 | 94.9 | 92.1 | 94.5 | 96.5 | 98.8 |
| LCM-sdxl | 81.5 | 85.5 | 78.0 | 68.6 | 62.3 | 50.9 | 81.1 | 96.2 | 94.1 | 95.9 | 98.0 | 98.7 |
| LCM-sdv1-5 | 92.3 | 90.9 | 79.7 | 79.0 | 76.2 | 93.7 | 79.6 | 97.1 | 96.3 | 97.3 | 97.8 | 98.4 |
| FLUX.1-sch | 56.0 | 54.4 | 72.9 | 56.3 | 50.4 | 53.1 | 68.4 | 85.2 | 89.4 | 90.0 | 97.5 | 98.3 |
| FLUX.1-dev | 57.4 | 54.1 | 75.9 | 56.3 | 49.1 | 50.7 | 70.7 | 86.1 | 90.5 | 91.3 | 97.6 | 98.7 |
| Avg. | 69.4 | 71.9 | 76.5 | 63.4 | 57.4 | 65.7 | 76.0 | 87.1 | 91.7 | 92.1 | 97.6 | **98.6** |

Table 3: Results on Co-Spy-Bench Cheng et al. (2025). Methods are trained on DRCT-2M/SDv1.4 following CO-SPY Cheng et al. (2025), which means the evaluation of cross-dataset transfer. The metric is Averaged Accuracy with 0.5 threshold. The compared methods are CNNSpot Wang et al. (2020), LNP Liu et al. (2022), UnivFD Ojha et al. (2023), DIRE Wang et al. (2023), FreqNet Tan et al. (2024a), NPR Tan et al. (2024b), DRCT Chen et al. (2024b), CO-SPY Cheng et al. (2025), GenSign Fei et al. (2026), IAPL Li et al. (2026), and LOTA Wang et al. (2025).

# 4 EXPERIMENT

## 4.1 DATASETS AND IMPLEMENTATION DETAILS

Experiments are conducted on the GenImage Zhu et al. (2023a), DRCT-2M Chen et al. (2024a), Co-Spy-Bench Cheng et al. (2025), and ForenSynths Wang et al. (2020) (in the Appendix) benchmarks. The evaluation metric is accuracy with 0.5 threshold.

For the proposed MP-MFD, we use a predefined set of periods, e.g., $\{6, 8, 12\}$, and inject the corresponding folded images into different stages of ResNet-50 He et al. (2016). Specifically, folded images with modulo 6, 8, and 12 are used as the input to the network, the input to Stage2 of the network, and the input to Stage3 of the network, respectively. The adapters are implemented as three $3 \times 3$ Conv-BN-ReLU blocks with appropriate stride when downsampling is needed. We fine-tune the entire network with JPEG compression as data augmentation, and using the Adam optimizer with a learning rate of $10^{-4}$. The total batch size is 64, and the input size is $256 \times 256$. The model is trained with PyTorch 1.13.0 on one NVIDIA RTX A6000 GPU for 30 epochs.

### 4.2 Main Results

**GenImage.** Table 1 reports the results on GenImage. Following Zhu et al. Zhu et al. (2023b), all methods are trained on GenImage/SDv1.4. MP-MFD achieves the best average accuracy of 99.2%, outperforming LOTA (98.6%), IAPL (96.7%), and GenSign (96.6%). In particular, MP-MFD reaches 98.5% on Midjourney, which is a substantial improvement over LOTA (91.6%). It also obtains 99.9% on SDv1.4 and SDv1.5, and around 99% on ADM, GLIDE, Wukong, VQDM, and BigGAN. These results demonstrate the effectiveness and generalization of MP-MFD on GenImage.

**DRCT-2M.** Table 2 summarizes the results on DRCT-2M. Following DRCT Chen et al. (2024b), methods are trained on DRCT-2M/SDv1.4. MP-MFD achieves the best average accuracy of 95.4%, surpassing LOTA (91.6%), IAPL (92.5%), and GenSign (91.5%).

**Co-Spy-Bench.** We also conduct the evaluation of cross-dataset transfer. Table 3 reports the results on Co-Spy-Bench with methods trained on DRCT-2M/SDv1.4, which is the CO-SPY protocol Cheng et al. (2025). MP-MFD achieves the remarkable best average accuracy of 98.6%, surpassing LOTA(97.6%), IAPL (92.1%), and GenSign (91.7%).

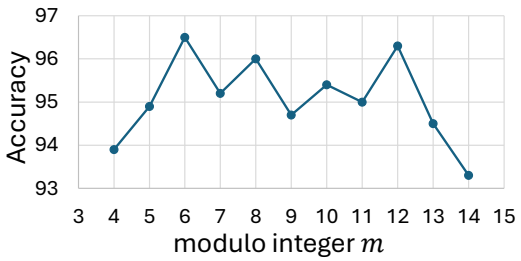


Figure 3: Different modulo integer period $m$ on GenImage. Only the single-period modulo folding image is used as the input to the network, in order to show the effectiveness of different modulo integer period $m$. Note that $m = 8$ is mathematically equivalent to the operation of combining the lowest three bit-planes in LOTA.

### 4.3 Ablation Study

**Effect of single modulo period.** Figure 3 studies single-period modulo folding on GenImage. Only the folded image from a single modulo period $m$ is used as network input. Note that $m = 8$ is mathematically equivalent to combining the lowest three bit-planes in LOTA. However, Figure 1 shows that $m = 8$ is not uniquely optimal. The best single-period accuracy is obtained at $m = 6$ (about 96.5%), while $m = 8$ and $m = 12$ also achieve strong results (about 96.0% and 96.3%, respectively). A broad range of periods from 4 to 14 remains effective, with $m = 14$ dropping to about 93.3%. This confirms that the essential mechanism is modulo folding, rather than the specific bit-plane choice.

**Multi-period and multi-stage injection.** Figure 3 shows that modulo 6, 8, 12, and 10 achieve the highest performance ranking in the top 4. Therefore, we use these moduli in Table 4 to study different settings of MP-MFD on GenImage. Using only modulo 6 as input yields 96.5%. Adding modulo 8 at Stage 2 improves accuracy to 98.3%. The best configuration is $\{6, 8, 12\}$, which injects modulo 6, 8, and 12 into the network input, Stage 2, and Stage 3, respectively, achieving 99.2%. Adding more periods, *e.g.*, modulo 10 at Stage 4 slightly reduces accuracy to 99.1%, suggesting that three periods are sufficient for capturing complementary artifact information. Table 5 further shows that the assignment order is not critical: $\{6, 8, 12\}$, $\{8, 12, 6\}$, $\{12, 6, 8\}$, and $\{6, 12, 8\}$ achieve 99.2%, 99.0%, 99.1%, and 99.1%, respectively. The best setting remains input $= 6$, Stage $2 = 8$, Stage $3 = 12$.

| Input | Stage2 | Stage3 | Stage4 | GenImage |
|---|---|---|---|---|
| 6 | | | | 96.5 |
| 6 | 8 | | | 98.3 |
| 6 | 8 | 12 | | **99.2** |
| 6 | 8 | 12 | 10 | 99.1 |

Table 4: Different settings of MP-MFD on the GenImage. For example, the third row means that the image with modulo 6, modulo 8, and modulo 12 are used as the input to the network, the input to Stage 2 of the network, and the input to Stage3 of the network, respectively. So does for the other rows.

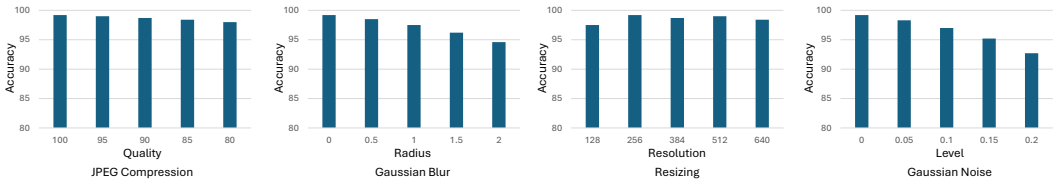


Figure 4: Robustness evaluation of MP-MFD on GenImage. Each subfigure denotes robustness evaluation against different transformation functions. In each subfigure, the x-axis denotes the varying parameters and the y-axis denotes the accuracy.

### 4.4 Robustness Evaluation

A key observation from our early experiments is that modulo-folded images are inherently fragile to image degradation. For example, JPEG compression perturbs low-order pixel values and can therefore drastically alter the folded representation, causing severe performance drops or even model collapse of the proposed MP-MFD. To address this, we deliberately include JPEG compression as data augmentation during training. This encourages the network to rely on the underlying artifact cues rather than memorizing fragile low-bit patterns. Figure 4 validates the effectiveness of this strategy. Specifically, we choose JPEG compression (*quality* = 100%,95%,90%,85%,80%), Gaussian blur ($\sigma$ = 0,0.5,1.0,1.5,2.0), Resizing (*resolution* = 128, 256, 384, 512, 640), and Gaussian Noise (*level* = 0.00, 0.05, 0.10, 0.15, 0.20) when testing. Figure 4 shows that MP-MFD achieves strong robustness to JPEG compression across different quality factors, mainly due to the data augmentation during training. Surprisingly, we find it generalizes well to degradations not seen during training, such as Gaussian blur, Resizing, and Gaussian noise, consistently preserving high accuracy. These results confirm that the proposed training scheme substantially improves the robustness of modulo-folding-based detection in practical settings. The robust experiments also show the limitation of MP-MFD, which is the extremely degraded images, such as too large noise level of Gaussian Noise shown in the Figure 4. Overall, MP-MFD maintains effective detection performance across the evaluated parameter ranges, showing that the modulo-folding representation preserves useful artifact cues under common image degradation.

| Input | Stage2 | Stage3 | GenImage |
|---|---|---|---|
| 6 | 8 | 12 | 99.2 |
| 8 | 12 | 6 | 99.0 |
| 12 | 6 | 8 | 99.1 |
| 6 | 12 | 8 | 99.1 |

Table 5: Different orders of the inputs to the MP-MFD on the GenImage. For example, the first row means that the image with modulo 6, modulo 8, and modulo 12 are used as the input to the network, the input to Stage 2 of the network, and the input to Stage3 of the network, respectively. So does for the other rows.

### 4.5 Inference Time

The inference time of MP-MFD and LOTA is 2.5 ms and 3.5 ms, respectively. The time cost of MP-MFD is mainly dominated by the forward process of backbone ResNet-50, since the time cost of modulo operation is negligible and the tiny adapter $\phi$ is very lightweight (only three $3 \times 3$ convs).

## 5 Conclusion

In this work, we reveal that LOTA's bit-plane operation is mathematically equivalent to modulo-8 folding, and that periodic modulo folding, rather than the specific bit-plane choice, is the essential mechanism for AI-generated image detection. We theoretically show that modulo folding compresses the semantic dynamic range while preserving small artifact perturbations, thereby amplifying artifact cues. Based on this, we propose MP-MFD, which fuses multiple modulo-folded images at different stages of a backbone network to capture complementary artifact cues without requiring reconstruction, iterative sampling, or generative inversion. Experiments on GenImage, DRCT-2M, Co-Spy-Bench, and ForenSynths (in the Appendix) show state-of-the-art performance. Our work unifies and extends bit-plane-based detection to a broader family of modulo operators.

## REFERENCES


George Cazenavette, Avneesh Sud, Thomas Leung, and Ben Usman. Fakeinversion: Learning to detect images from unseen text-to-image models by inverting stable diffusion. In *CVPR*, pp. 10759–10769, 2024.

Lucy Chai, David Bau, Ser-Nam Lim, and Phillip Isola. What makes fake images detectable? understanding properties that generalize. In *European conference on computer vision*, pp. 103–120. Springer, 2020.

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*, 2024a.

Baoying Chen, Jishen Zeng, Jianquan Yang, and Rui Yang. Drct: Diffusion reconstruction contrastive training towards universal detection of diffusion generated images. In *ICML*, 2024b.

Jiaxuan Chen, Jieteng Yao, and Li Niu. A single simple patch is all you need for ai-generated image detection. *arXiv preprint arXiv:2402.01123*, 2024c.

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

Sungik Choi, Hankook Lee, Jaehoon Lee, Robin Kim, Stanley Jungkyu Choi, and Moontae Lee. A debiased reconstruction-based framework for training-free detection of ai-generated images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3254–3263, 2026.

Beilin Chu, Xuan Xu, Xin Wang, Yufei Zhang, Weike You, and Linna Zhou. Fire: Robust detection of diffusion-generated images via frequency-guided reconstruction error. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 12830–12839, 2025.

Riccardo Corvi, Davide Cozzolino, Giada Zingarini, Giovanni Poggi, Koki Nagano, and Luisa Verdoliva. On the detection of synthetic images generated by diffusion models. In *ICASSP*, pp. 1–5. IEEE, 2023.

Jianwei Fei, Yunshu Dai, Xiaoyu Zhou, Zhihua Xia, and Alessandro Piva. Enabling supervised learning of generative signatures for generalized ai-generated images detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14084–14094, 2026.

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

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Yonghyun Jeong, Doyeon Kim, Seungjai Min, Seongho Joe, Youngjune Gwon, and Jongwon Choi. Bihpf: Bilateral high-pass filters for robust deepfake detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 48–57, 2022a.

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

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

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*, pp. 21262–21272, 2026.

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

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*, pp. 10770–10780, 2024.

Zhengzhe Liu, Xiaojuan Qi, and Philip HS Torr. Global texture enhancement for fake face detection in the wild. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8060–8069, 2020.

Yuchen Luo, Yong Zhang, Junchi Yan, and Wei Liu. Generalizing face forgery detection with high-frequency features. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 16317–16326, 2021.

Yunpeng Luo, Junlong Du, Ke Yan, and Shouhong Ding. Lareˆ 2: Latent reconstruction error based method for diffusion-generated image detection. In *CVPR*, pp. 17006–17015, 2024.

Ruipeng Ma, Jinhao Duan, Fei Kong, Xiaoshuang Shi, and Kaidi Xu. Exposing the fake: Effective diffusion-generated images detection. *arXiv preprint arXiv:2307.06272*, 2023.

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

Yuyang Qian, Guojun Yin, Lu Sheng, Zixuan Chen, and Jing Shao. *Thinking in Frequency: Face Forgery Detection by Mining Frequency-aware Clues*, pp. 86–103. Jan 2020a.

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

Jonas Ricker, Simon Damm, Thorsten Holz, and Asja Fischer. Towards the detection of diffusion model deepfakes. *arXiv preprint arXiv:2210.14571*, 2022.

Jonas Ricker, Denis Lukovnikov, and Asja Fischer. Aeroblade: Training-free detection of latent diffusion images using autoencoder reconstruction error. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9130–9140, 2024.

Zeyang Sha, Zheng Li, Ning Yu, and Yang Zhang. De-fake: Detection and attribution of fake images generated by text-to-image generation models. In *Proceedings of the 2023 ACM SIGSAC conference on computer and communications security*, pp. 3418–3432, 2023.

Chuangchuang Tan, Yao Zhao, Shikui Wei, Guanghua Gu, and Yunchao Wei. Learning on gradients: Generalized artifacts representation for gan-generated images detection. In *CVPR*, 2023.

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*, volume 38, pp. 5052–5060, 2024a.

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 *CVPR*, pp. 28130–28139, 2024b.

Hongsong Wang, Renxi Cheng, Yang Zhang, Chaolei Han, and Jie Gui. Lota: Bit-planes guided ai-generated image detection. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 17246–17255. IEEE, 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 *CVPR*, pp. 8695–8704, 2020.

Zhendong Wang, Jianmin Bao, Wengang Zhou, Weilun Wang, Hezhen Hu, Hong Chen, and Houqiang Li. Dire for diffusion-generated image detection. In *CVPR*, pp. 22445–22455, 2023.

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*, 2025.

Lin Yuan, Xiaowan Li, Yan Zhang, Jiawei Zhang, Hongbo Li, and Xinbo Gao. Mlep: Multi-granularity local entropy patterns for generalized ai-generated image detection. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

Chende Zheng, Chenhao Lin, Zhengyu Zhao, Hang Wang, Xu Guo, Shuai Liu, and Chao Shen. Breaking semantic artifacts for generalized ai-generated image detection. *Advances in Neural Information Processing Systems*, 37:59570–59596, 2024.

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

Nan Zhong, Haoyu Chen, Yiran Xu, Zhenxing Qian, and Xinpeng Zhang. Beyond generation: A diffusion-based low-level feature extractor for detecting ai-generated images. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 8258–8268, 2025.

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

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. *NeurIPS*, 2023b.

# A Appendix

## A.1 More Dataset Details

Experiments are conducted on the following benchmarks: GenImage Zhu et al. (2023a), DRCT-2M Chen et al. (2024a), Co-Spy-Bench Cheng et al. (2025), and ForenSynths dataset Wang et al. (2020). The training set of GenImage contains fake images generated using Stable Diffusion V1.4. For the test set, we incorporate a variety of generators, including Stable Diffusion V1.4, Stable Diffusion V1.5, GLIDE, VQDM, Wukong, BigGAN, ADM, and Midjourney. DRCT-2M includes 16 types of generated images, with 120k images for each type: 10 types of the currently available SD text-to-image models, 6 types of ControlNet and 3 types diffusion inpainting models. Co-Spy-Bench consists of 22 types text-to-image generative models: 8 types Stable Diffusion variants, 4 types Playground models, 4 types Segmind models, 2 types PixArt models, 2 types LCM models, and 2 types Flux models. ForenSynths contains 7 types of GAN-based generated images (ProGAN, StyleGAN, StyleGAN2, BigGAN, CycleGAN, StarGAN, GauGAN) and 1 Deepfake forgery images.

## A.2 GAN-generated Images

We compare the proposed MP-MFD on the ForenSynths dataset Wang et al. (2020). As shown in Table 6, the proposed MP-MFD also achieves state-of-the-art results on GAN-generated images, when compared with other state-of-the-art methods, such as FatFormer Liu et al. (2024) and CoD Jia et al. (2025). Specifically, the proposed MP-MFD achieves a mean accuracy of 99.0%, demonstrating its effectiveness in GAN-generated image detection.

## A.3 More Ablation Study

Table 7 studies the effect of different implementations of the tiny network $\phi$. When the number of blocks is increased from one to three, the accuracy improves from 98.1% to 99.2%, indicating that a moderately deeper $\phi$ provides stronger feature representation and helps extract more discriminative

| Methods | ProGAN | StyleGAN | StyleGAN2 | BigGAN | CycleGAN | StarGAN | GauGAN | Deepfake | Mean |
|---|---|---|---|---|---|---|---|---|---|
| CNNSpot Wang et al. (2020) | 91.4 | 63.8 | 76.4 | 52.9 | 72.7 | 63.8 | 63.9 | 51.7 | 67.1 |
| FreqFD Frank et al. (2020) | 90.3 | 74.5 | 73.1 | 88.7 | 75.5 | 99.5 | 69.2 | 60.7 | 78.9 |
| PatchFor Chai et al. (2020) | 97.8 | 82.6 | 83.6 | 64.7 | 74.5 | 100.0 | 57.2 | 85.0 | 80.7 |
| F3Net Qian et al. (2020b) | 99.4 | 92.6 | 88.0 | 65.3 | 76.4 | 100.0 | 58.1 | 63.5 | 80.4 |
| BiHPF Jeong et al. (2022a) | 90.7 | 76.9 | 76.2 | 84.9 | 81.9 | 94.4 | 69.5 | 54.4 | 78.6 |
| FrePGAN Jeong et al. (2022b) | 99.0 | 80.7 | 84.1 | 69.2 | 71.1 | 99.9 | 60.3 | 70.9 | 79.4 |
| LGrad Tan et al. (2023) | 99.9 | 94.8 | 96.0 | 82.9 | 85.3 | 99.6 | 72.4 | 58.0 | 86.1 |
| UnivFD Ojha et al. (2023) | 99.7 | 89.0 | 83.9 | 90.5 | 87.9 | 91.4 | 89.9 | 80.2 | 89.1 |
| NPR Tan et al. (2024b) | 99.8 | 96.3 | 97.3 | 87.5 | 95.0 | 99.7 | 86.6 | 77.4 | 92.5 |
| FatFormer Liu et al. (2024) | 99.9 | 97.2 | 98.8 | 99.5 | 99.3 | 99.8 | 99.4 | 93.2 | 98.4 |
| CoD Jia et al. (2025) | 99.9 | 98.2 | 98.6 | 93.6 | 99.1 | 99.9 | 99.9 | 93.4 | 98.2 |
| MP-MFD | 99.9 | 99.3 | 99.2 | 99.6 | 99.4 | 99.9 | 99.4 | 95.3 | **99.0** |

Table 6: Accuracy comparisons with state-of-the-art methods on ForenSynths dataset. The decision threshold is 0.5.

| $\phi$ | GenImage |
|---|---|
| one $3 \times 3$ Conv-BN-ReLU blocks | 98.1 |
| two $3 \times 3$ Conv-BN-ReLU blocks | 98.9 |
| three $3 \times 3$ Conv-BN-ReLU blocks | 99.2 |
| four $3 \times 3$ Conv-BN-ReLU blocks | 99.1 |

Table 7: Different implementation of the tiny network $\phi$.

information. However, further increasing the depth to four blocks slightly reduces the accuracy to 99.1%, suggesting that the performance gain from additional depth has saturated. Therefore, we adopt three Conv-BN-ReLU blocks as the default implementation of $\phi$.