# CHROMAGUARD: SIMPLE CHROMA, STRONG GENERALIZATION FOR SYNTHETIC IMAGE DETECTION

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

## ABSTRACT

Pretrained models have enabled remarkable performance in synthetic image detection by providing rich and transferable visual representations. However, their semantic information can become spuriously associated with forgery labels, causing detectors to rely on semantic shortcuts rather than synthesis artifacts and limiting generalization. To address this challenge, we propose **ChromaGuard**, a framework for generalizable synthetic image detection that employs chroma guidance to reduce semantic reliance. Through visualization analysis, we reveal that YCbCr chroma is very weakly related to image semantics. Building on this observation, ChromaGuard treats the RGB image and its chroma component as complementary views. The RGB branch preserves the complete image information to avoid missing synthesis artifacts, while the chroma branch guides the detector toward cues weakly associated with semantic content. A simple bidirectional Chroma-RGB interaction module connects the two branches, which alternately injects chroma guidance into RGB features and refines chroma features using the updated RGB representation. To further enhance sensitivity to synthesis artifacts within RBG branch, we introduce a DCT high-frequency adapter to mine frequency-domain artifacts left by generators. Evaluations of cross-generator generalization, semantic sensitivity, and robustness demonstrate its strong performance and reliability, supporting the effectiveness of Chroma guidance.

## 1 INTRODUCTION

Recent advances in generative adversarial networks (GANs) Goodfellow et al. (2020); Karras et al. (2018; 2019) and diffusion models Ho et al. (2020); Nichol & Dhariwal (2021); Rombach et al. (2022) have made synthetic images increasingly difficult to distinguish from real ones. To generalize across generators, detectors need to recognize synthesis artifacts that are less dependent on image content. Early approaches primarily learned spatial-domain features to capture synthesis artifacts Wang et al. (2020). Subsequent methods investigated frequency-domain artifacts that complement spatial cues Frank et al. (2020); Jeong et al. (2022); Tan et al. (2024). More recently, CLIP-based detectors have improved cross-generator generalization by using the rich, transferable visual representations learned during large-scale pretraining Liu et al. (2024); Ojha et al. (2023); Fu et al. (2025); Mu et al. (2026).

However, CLIP Radford et al. (2021) is pretrained on large-scale image–text pairs, so its representations primarily encode high-level semantics such as object identity and scene layout. These attributes do not determine whether an image is real or synthetic, but they may correlate spuriously with forgery labels during training Zhang et al. (2025); Yang et al. (2026a); Liu et al. (2026). A detector can then achieve high in-distribution accuracy by exploiting these semantic shortcuts rather than recognizing synthesis artifacts. These shortcuts lead to inaccurate decision boundaries by preventing detectors from learning to recognize synthesis artifacts. The challenge is therefore to reduce semantic reliance while retaining the transferability of pretrained models.

Recent methods address this issue by imposing feature-level constraints that reduce reliance on semantic cues Zhang et al. (2025); Tao et al. (2025); Yang et al. (2026a); Liu et al. (2026). However, these constraints operate on features that already encode rich semantic information, and thus yield only marginal gains in generalization over existing methods. This motivates us to seek a form of guidance that, at an earlier stage, steers the detector to place greater emphasis on weakly semanti-



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

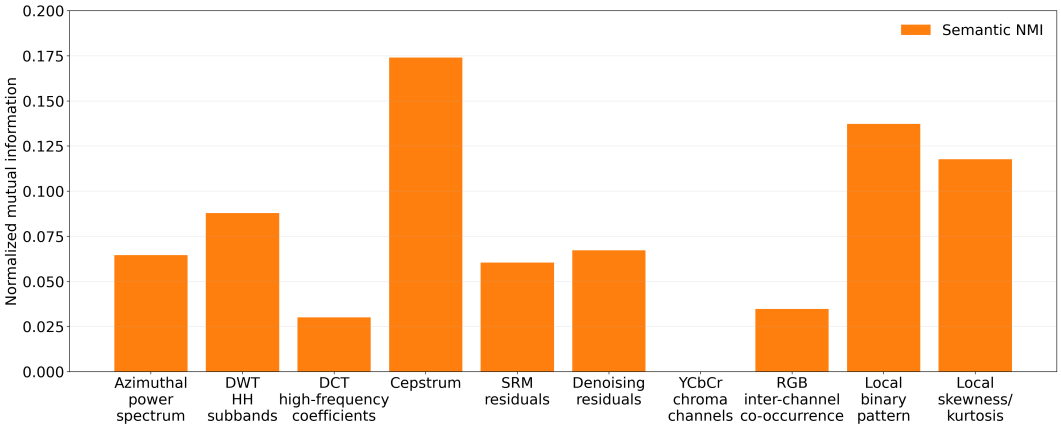



Figure 1: Estimated normalized mutual information (NMI) between various image-level components and semantic labels, which serve as a proxy for semantic content. Among all components, the YCbCr chroma channels yield the lowest estimated NMI, with a value close to zero, motivating their use as a guidance view with weak semantic association.

cally relevant cues. To this end, we turn to the image level, where guidance can be applied from the very outset of feature extraction, progressively steering the detector to place greater emphasis on weakly semantically relevant cues. Specifically, we estimate the normalized mutual information (NMI) between common image-level components and semantic labels, which serve as a proxy for semantic content. A smaller NMI indicates weaker semantic relevance of a component. The analysis is conducted on the training set of AIGCDetectBenchmark Zhong et al. (2023), which consists of 20 semantic categories. As shown in Fig. 1, the YCbCr chroma channels exhibit the lowest estimated NMI among the components we evaluate, with a value close to zero. This observation reveals their potential to bias the detector toward weakly semantically relevant features.

Building on this observation, we propose **ChromaGuard**, a framework for synthetic image detection that employs chroma guidance to reduce semantic reliance and improve generalization. ChromaGuard treats the RGB image and its chroma components as complementary views. The RGB branch preserves the complete image information to avoid discarding synthesis artifacts, while the chroma branch guides the detector toward cures weakly associated with semantic content. A simple bidirectional Chroma-RGB interaction module connects the two branches, which alternately injects chroma guidance into RGB features and refines chroma features using the updated RGB representation. This interaction encourages the detector to learn features that depend less on semantic content. To further enhance sensitivity to synthesis artifacts, we introduce a DCT high-frequency adapter within the RGB branch. This design is motivated by two factors: (i) the well-established finding that generative models leave distinctive frequency artifacts Tan et al. (2024); Qian et al. (2020); Li et al. (2021); Wang et al. (2020); and (ii) our complementary observation that DCT high-frequency components carry weak semantic relevance, so that the adapter does not disrupt the features learned by the interaction module. Together, these components encourage ChromaGuard to distinguish real from synthetic images based on synthesis artifacts rather than semantic content.

Our main contributions can be summarized as follows:

- We investigate semantic reliance from an image-level perspective. The visualization analysis shows that the YCbCr chroma channels exhibit minimal semantic association, indicating their utility as a guidance view for forensic representation learning.
- We propose ChromaGuard, which employs a bidirectional Chroma-RGB interaction module to steer the detector toward weakly semantically relevant cues, and further incorporates a DCT high-frequency adapter within the RGB branch to enhance its sensitivity to artifacts.
- We evaluate ChromaGuard under three complementary protocols assessing generalization, semantic sensitivity, and robustness. ChromaGuard consistently achieves the best performance among all compared methods, demonstrating the effectiveness of chroma as a guidance signal only weakly associated with semantic content.

## 2 Related Work

Early synthetic-image detectors primarily relied on spatial-domain Li et al. (2020); Shiohara & Yamasaki (2022); Jiang et al. (2024); Zhang et al. (2026); Yan et al. (2025a) or frequency-domain Frank et al. (2020); Qian et al. (2020); Jeong et al. (2022); Tan et al. (2024); Wang et al. (2026) artifacts introduced during image synthesis. As generators become more diverse and produce more realistic images, detection methods increasingly use large-scale pretrained models, particularly CLIP Radford et al. (2021), to improve cross-generator generalization. UnivFD Ojha et al. (2023) directly uses frozen features to detect synthetic images and demonstrates promising transferability across generation methods. FatFormer Liu et al. (2024) adapts CLIP with a forgery-aware adapter that incorporates spatial and frequency-domain cues to capture subtle synthesis artifacts. Rine Koutlis & Papadopoulos (2024) aggregates features from different layers of CLIP to achieve generalizable detection. LTD Yang et al. (2026b) exploits latent transition discrepancies to detect synthetic images. However, CLIP is pretrained on large-scale image-text pairs, so its representations encode rich semantic information that may correlate spuriously with forgery labels during training Zhang et al. (2025); Yang et al. (2026a); Liu et al. (2026).

Recent studies impose feature-level constraints to reduce reliance on semantic cues. VIB-Net Zhang et al. (2025) employs a Variational Information Bottleneck to suppress task-irrelevant semantic information. SAGNet Tao et al. (2025) applies gradient reversal to the category-related loss to suppress semantic-specific information. CausalCLIP Liu et al. (2026) separates CLIP features into causal and non-causal subspaces and filters the latter to suppress spurious correlations between forgery labels and semantic content. The HSIC Bottleneck Yang et al. (2026a) applies an information bottleneck loss to intermediate CLIP ViT features to suppress semantic information. However, these constraints operate on features that already encode rich semantic information, yielding only marginal gains in generalization. To this end, we turn to the image level, where guidance can be applied from the very outset of feature extraction, progressively steering the detector to place greater emphasis on weakly semantically relevant cues. Our visualizations show that YCbCr chroma is only weakly associated with semantic content. Based on this observation, we propose ChromaGuard, a chroma-guided framework for generalizable synthetic image detection.

## 3 Methodology

### 3.1 Overview

Figure 2 provides an overview of **ChromaGuard**. It uses a simple bidirectional Chroma-RGB interaction module to guide the detector toward components weakly associated with semantic content. A DCT high-frequency adapter in the RGB branch further enhances sensitivity to synthesis artifacts.

### 3.2 Bidirectional Chroma-RGB interaction module

Figure 1 shows that YCbCr chroma is only weakly associated with semantic content, motivating its use as guidance to reduce semantic reliance. To this end, we adopt a simple bidirectional interaction module. Specifically, we convert the input RGB image to YCbCr and extract the Cb and Cr components. We form a single-channel chroma map by computing the Euclidean magnitude $C(x, y) = \sqrt{C_b(x, y)^2 + C_r(x, y)^2}$ at each pixel. This operation preserves spatial variations in chroma magnitude while discarding chroma direction, reducing reliance on category-specific color cues. The chroma map is processed by a three-layer ConvBlock to obtain chroma features $\mathbf{F}_c$. In parallel, the input RGB image is fed into the frozen CLIP visual encoder to obtain RGB features $\mathbf{F}_r$.

Before the first ViT block in each interaction module, we use the RGB features as queries and the chroma features as keys and values. Cross-attention $\mathcal{CA}$ injects chroma guidance into the RGB features to emphasize cues weakly associated with semantic content:

$$\widetilde{\mathbf{F}}_r = \mathbf{F}_r + \mathcal{CA}(\mathbf{F}_r, \mathbf{F}_c, \mathbf{F}_c). \tag{1}$$

The updated RGB features $\widetilde{\mathbf{F}}_r$ are then passed through subsequent CLIP Transformer blocks. For simplicity, we also denote the resulting RGB features by $\widetilde{\mathbf{F}}_r$. After the last ViT block in the interaction module, we refine the chroma features through cross-attention, using chroma features as queries

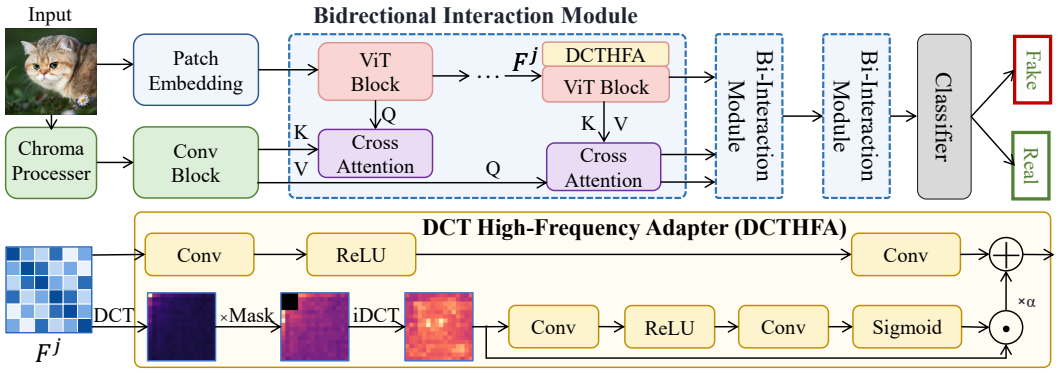



Figure 2: Overview of the proposed framework. It integrates RGB and chroma features through bidirectional interaction and enhances artifact sensitivity with a DCT high-frequency adapter.

and the resulting RGB features as keys and values. This process can be formulated as follows:

$$\widetilde{\mathbf{F}}_c = \mathbf{F}_c + \mathcal{CA}(\mathbf{F}_c, \widetilde{\mathbf{F}}_r, \widetilde{\mathbf{F}}_r). \tag{2}$$

The updated chroma features are passed to the next interaction stage. From the outset of feature extraction, this alternating interaction progressively guides the detector to learn features weakly associated with semantic content.

### 3.3 DCT High-Frequency Adapter

Generative models leave distinctive frequency-domain artifacts Tan et al. (2024); Qian et al. (2020); Li et al. (2021); Wang et al. (2020). Our analysis also shows that DCT high-frequency components are weakly associated with semantic content. These findings motivate a DCT high-frequency adapter in the RGB branch to enhance sensitivity to synthesis artifacts while preserving the weak semantic association of the features learned through chroma guidance.

The adapter consists of a spatial processor and a DCT processor. Let $\mathbf{F}^j \in \mathbb{R}^{B\times N\times d}$ denote the input to the last ViT block in the $j$-th bidirectional interaction stage. Here, $B$, $N$, and $d$ denote the batch size, token count, and feature dimension, respectively. The $N$ tokens include one class token and the patch tokens. The adapter takes $\mathbf{F}^j$ as input and operates in parallel with the last ViT block. The spatial processor is a bottleneck MLP implemented with two $1 \times 1$ convolutional layers. It projects $\mathbf{F}^j$ to a bottleneck dimension $d_b \ll d$, applies ReLU and dropout, and projects the features back to dimension $d$. This yields the spatial adaptation $\widehat{\mathbf{F}}^j_{\text{spa}}$.

In the DCT processor, we first remove the class token from $\mathbf{F}^j$ and reshape the remaining patch tokens into a two-dimensional feature map $\mathbf{F}^j_p$. We then apply a 2D DCT independently to each channel over the full $16 \times 16$ spatial grid. A fixed binary mask $\mathbf{M}$ suppresses the upper-left $4 \times 4$ region, corresponding to a masking ratio of 0.25 along each spatial frequency axis. Specifically, $M_{u,v} = 0$ when $0 \le u < 4$ and $0 \le v < 4$, and $M_{u,v} = 1$ otherwise. Subsequently, we apply the corresponding inverse DCT and multiply the reconstructed features element-wise with the original feature map to obtain modulated features:

$$\mathbf{F}^j_{\text{freq}} = \text{IDCT}\left(\mathbf{M} \odot \text{DCT}\left(\mathbf{F}^j_p\right)\right) \odot \mathbf{F}^j_p, \tag{3}$$

where $\odot$ denotes element-wise multiplication. An importance-recalibration module consisting of a convolutional layer and a sigmoid activation is then applied to emphasize informative frequency responses. The resulting feature map is reshaped into patch tokens. We insert a zero vector at the class-token position to obtain the frequency adaptation $\widehat{\mathbf{F}}^j_{\text{freq}}$.

Finally, the spatial and frequency adaptations are combined with the output feature $\mathbf{F}^j_o$ of the last ViT block in the bidirectional interaction module to obtain the final output:

$$\mathbf{F}^j_{\text{out}} = \mathbf{F}^j_o + \lambda(\widehat{\mathbf{F}}^j_{\text{spa}} + \widehat{\mathbf{F}}^j_{\text{freq}}), \tag{4}$$

where $\lambda$ is a learnable scale factor that controls the contribution of the combined spatial and frequency adaptations.

Table 1: Accuracy (%) and reliability (Rel.) for **generalization evaluation**. The last two columns report statistics computed jointly across all generators. Most baseline results are from SimLBR Dhakal et al. (2026). **Bold** and underlined values mark the best and second-best results.

**Panel A: GAN-based Generators**

| Method | Type | ProGAN | StyleGAN | BigGAN | CycleGAN | StarGAN | GauGAN | StyleGAN2 | WFIR |
|---|---|---|---|---|---|---|---|---|---|
| CNNSpot | Spat | 100.0 | 90.17 | 71.17 | 87.62 | 94.60 | 81.42 | 86.91 | 91.65 |
| FreDect | Freq | 99.36 | 78.02 | 81.97 | 78.77 | 94.62 | 80.57 | 66.19 | 50.75 |
| Fusing | Spat | 100.0 | 85.20 | 77.40 | 87.00 | 97.00 | 77.00 | 83.30 | 66.80 |
| LNP | Spat | 99.67 | 91.75 | 77.75 | 84.10 | 99.92 | 75.39 | 94.64 | 70.85 |
| LGrad | Grad | 99.83 | 91.08 | 85.62 | 86.94 | 99.27 | 78.46 | 85.32 | 55.70 |
| UnivFD | Spat | 99.81 | 84.93 | 95.08 | 98.33 | 95.75 | 99.47 | 74.96 | 86.90 |
| DIRE-G | RecE | 95.19 | 83.03 | 70.12 | 74.19 | 95.47 | 67.79 | 75.31 | 58.05 |
| DIRE-D | RecE | 52.75 | 51.31 | 49.70 | 49.58 | 46.72 | 51.23 | 51.72 | 53.30 |
| PatchCraft | Spat | 100.0 | 92.77 | 95.80 | 70.17 | 99.97 | 71.58 | 89.55 | 85.80 |
| NPR | PixR | 99.79 | 97.70 | 84.35 | 96.10 | 99.35 | 82.50 | 98.38 | 65.80 |
| Rine | Spat | 100.0 | 97.33 | 94.55 | 96.48 | 100.0 | 83.24 | 96.22 | 93.85 |
| FatFormer | Spat | 100.0 | 93.37 | 99.18 | 98.52 | 99.85 | 99.41 | 94.49 | 94.65 |
| Effort | Spat | 100.0 | 95.38 | 99.20 | 99.77 | 99.92 | 99.54 | 96.64 | 78.90 |
| AIDE | Freq | 99.99 | 99.64 | 83.95 | 98.48 | 99.91 | 73.25 | 98.00 | 94.20 |
| MoeFFD | Spat | 99.73 | 88.82 | 82.13 | 97.50 | 92.82 | 92.38 | 89.03 | 81.50 |
| HSIC | Spat | 100.0 | 91.22 | 82.55 | 95.84 | 100.0 | 69.84 | 94.07 | 98.05 |
| SimLBR | Spat | 99.29 | 87.38 | 95.52 | 88.11 | 88.54 | 97.14 | 84.57 | 89.50 |
| **Ours** | Spat | 100.0 | 98.89 | 93.78 | 98.18 | 99.80 | 92.08 | 98.98 | 98.75 |

**Panel B: DM-based Generators**

| Method | ADM | Glide | Midjourney | SDv1.4 | SDv1.5 | VQDM | Wukong | DALLE2 | *Mean*↑ | *Rel.*↑ |
|---|---|---|---|---|---|---|---|---|---|---|
| CNNSpot | 60.39 | 58.07 | 51.39 | 50.57 | 50.53 | 56.46 | 51.03 | 50.45 | 70.78 | 1.10 |
| FreDect | 63.42 | 54.13 | 45.87 | 38.79 | 39.21 | 77.80 | 40.30 | 34.71 | 64.03 | 0.67 |
| Fusing | 49.00 | 57.20 | 52.20 | 51.00 | 51.40 | 55.10 | 51.70 | 52.80 | 68.38 | 1.02 |
| LNP | 84.73 | 80.52 | 65.55 | 85.55 | 85.67 | 74.46 | 82.06 | 88.75 | 83.84 | 3.46 |
| LGrad | 67.15 | 66.11 | 65.35 | 63.02 | 63.67 | 72.99 | 59.55 | 65.45 | 75.34 | 1.78 |
| UnivFD | 66.87 | 62.46 | 56.13 | 63.66 | 63.49 | 85.31 | 70.93 | 50.75 | 78.43 | 1.70 |
| DIRE-G | 75.78 | 71.75 | 58.01 | 49.74 | 49.83 | 53.68 | 54.46 | 66.48 | 68.68 | 1.29 |
| DIRE-D | 98.25 | 92.42 | 89.45 | 91.24 | 91.63 | 91.90 | 90.90 | 92.45 | 71.53 | 1.00 |
| PatchCraft | 82.17 | 83.79 | 90.12 | 95.38 | 95.30 | 88.91 | 91.07 | 96.60 | 89.31 | 4.42 |
| NPR | 69.69 | 78.36 | 77.85 | 78.63 | 78.89 | 78.13 | 76.11 | 64.90 | 82.91 | 2.76 |
| Rine | 89.52 | 95.72 | 61.38 | 97.62 | 97.70 | 94.63 | 96.56 | 85.10 | 92.49 | 4.44 |
| FatFormer | 79.17 | 84.24 | 56.90 | 88.18 | 87.14 | 90.03 | 90.84 | 57.60 | 88.35 | 2.83 |
| Effort | 75.06 | 92.68 | 54.18 | 88.23 | 87.96 | 85.58 | 85.44 | 78.10 | 88.54 | 3.09 |
| AIDE | 93.43 | 95.09 | 77.20 | 93.00 | 92.85 | 95.16 | 93.55 | 96.60 | 92.77 | 5.40 |
| MoeFFD | 58.43 | 62.61 | 52.23 | 63.01 | 64.04 | 66.21 | 61.09 | 53.25 | 75.30 | 1.52 |
| HSIC | 92.35 | 96.93 | 66.93 | 98.69 | 98.64 | 97.70 | 98.31 | 97.45 | 92.41 | 4.10 |
| SimLBR | 85.89 | 90.88 | 75.54 | 85.51 | 84.43 | 93.20 | 90.08 | 78.80 | 88.40 | 6.16 |
| **Ours** | 92.88 | 95.49 | 77.93 | 98.92 | 98.59 | 98.06 | 98.78 | 93.40 | **95.91** | **8.38** |

### 3.4 Training Objective

After three bidirectional interaction stages, we obtain an image representation $\mathbf{F}_{out} \in \mathbb{R}^{B\times d}$ for classification from the final module output. A lightweight classification head $\mathcal{C}$ maps $\mathbf{F}_{out}$ to a logit for each image. We optimize the trainable components using the binary cross-entropy loss:

$$\mathcal{L}_{\text{CE}} = -\big[\, y \log \sigma(\mathcal{C}(\mathbf{F}_{out})) + (1-y)\log(1-\sigma(\mathcal{C}(\mathbf{F}_{out})))\,\big], \tag{5}$$

where $y \in \{0, 1\}$ is the ground-truth label of the input image and $\sigma$ denotes the Sigmoid function.

Table 2: Accuracy (%) under three training settings for **semantic sensitivity evaluation**. Here, $\#n$ denotes the number of semantic categories used for training, with values of 1, 2, or 4. The corresponding average precision (AP) comparison results are shown in the **Appendix**.

| Methods | #n | ProGAN | StyleGAN | StyleGAN2 | BigGAN | CycleGAN | StarGAN | GauGAN | Deepfake | *Mean*↑ |
|---|---|---|---|---|---|---|---|---|---|---|
| CNNSpot | 1 | 95.2 | 79.2 | 73.8 | 62.9 | 49.6 | 94.9 | 57.2 | 82.9 | 74.5 |
| F3Net | 1 | 99.6 | 97.0 | 91.5 | 61.6 | 47.3 | 100.0 | 49.5 | 72.5 | 77.4 |
| SelfBlend | 1 | 86.6 | 55.7 | 54.5 | 55.5 | 72.3 | 58.0 | 82.3 | 53.0 | 64.7 |
| UnivFD | 1 | 99.0 | 88.2 | 71.4 | 90.5 | 96.9 | 96.4 | 90.3 | 70.3 | 87.9 |
| FatFormer | 1 | 97.8 | 75.8 | 73.1 | 88.7 | 98.5 | 67.4 | 99.5 | 52.6 | 81.7 |
| FreqNet | 1 | 98.0 | 92.0 | 89.5 | 85.5 | 96.1 | 94.2 | 91.8 | 69.8 | 89.6 |
| NPR | 1 | 95.8 | 95.2 | 87.0 | 72.5 | 71.9 | 99.7 | 54.8 | 79.1 | 82.0 |
| Rine | 1 | 100.0 | 96.6 | 88.3 | 96.8 | 97.0 | 97.8 | 95.9 | 86.5 | <u>94.9</u> |
| Effort | 1 | 95.9 | 79.6 | 68.6 | 93.4 | 99.6 | 99.0 | 98.8 | 76.4 | 88.9 |
| MoeFFD | 1 | 97.3 | 87.7 | 77.5 | 75.2 | 97.3 | 76.7 | 84.6 | 70.4 | 83.3 |
| SimLBR | 1 | 92.6 | 78.8 | 79.7 | 89.5 | 98.5 | 81.5 | 90.9 | 68.4 | 85.0 |
| **Ours** | 1 | 99.9 | 96.9 | 89.9 | 97.5 | 99.6 | 99.8 | 99.5 | 92.0 | **96.9** |
| CNNSpot | 2 | 98.3 | 69.6 | 61.7 | 67.8 | 61.5 | 93.7 | 69.9 | 82.6 | 75.7 |
| F3Net | 2 | 97.8 | 95.6 | 93.0 | 62.9 | 62.9 | 99.9 | 51.7 | 76.8 | 80.0 |
| SelfBlend | 2 | 87.5 | 54.1 | 55.9 | 55.8 | 78.8 | 60.3 | 85.3 | 56.8 | 66.8 |
| UnivFD | 2 | 98.9 | 82.6 | 68.6 | 91.3 | 95.8 | 97.2 | 93.2 | 80.3 | 88.5 |
| FatFormer | 2 | 99.7 | 87.1 | 81.7 | 97.1 | 100.0 | 98.5 | 99.7 | 86.3 | 93.8 |
| FreqNet | 2 | 99.6 | 90.4 | 85.8 | 89.0 | 96.7 | 97.5 | 88.0 | 80.7 | 91.0 |
| NPR | 2 | 97.6 | 96.1 | 89.3 | 77.3 | 90.0 | 100.0 | 63.9 | 65.1 | 84.9 |
| Rine | 2 | 100.0 | 93.3 | 81.3 | 98.5 | 99.6 | 99.8 | 99.1 | 86.1 | <u>94.7</u> |
| Effort | 2 | 99.0 | 86.2 | 79.6 | 98.4 | 100.0 | 99.9 | 99.6 | 73.8 | 92.1 |
| MoeFFD | 2 | 99.1 | 84.3 | 77.8 | 79.9 | 98.5 | 87.7 | 87.0 | 63.9 | 84.8 |
| SimLBR | 2 | 97.0 | 81.2 | 79.6 | 92.5 | 93.8 | 83.4 | 95.4 | 62.9 | 85.7 |
| **Ours** | 2 | 100.0 | 99.8 | 94.1 | 98.4 | 99.6 | 99.2 | 99.1 | 83.3 | **96.7** |
| CNNSpot | 4 | 99.7 | 92.0 | 80.1 | 66.0 | 58.1 | 99.0 | 65.0 | 89.7 | 81.2 |
| F3Net | 4 | 99.6 | 97.0 | 94.8 | 72.9 | 69.6 | 100.0 | 59.3 | 79.4 | 84.1 |
| SelfBlend | 4 | 97.4 | 69.3 | 62.8 | 58.8 | 84.2 | 58.9 | 83.5 | 53.5 | 71.0 |
| UnivFD | 4 | 99.8 | 94.7 | 79.0 | 92.5 | 94.6 | 97.0 | 94.4 | 83.1 | 91.9 |
| FatFormer | 4 | 100.0 | 98.2 | 86.7 | 97.8 | 100.0 | 97.7 | 99.8 | 68.5 | 93.6 |
| FreqNet | 4 | 99.6 | 90.2 | 88.0 | 90.5 | 95.8 | 85.7 | 93.4 | 88.9 | 91.5 |
| NPR | 4 | 99.4 | 98.6 | 95.4 | 80.3 | 89.2 | 100.0 | 67.8 | 59.7 | 86.3 |
| Rine | 4 | 99.0 | 98.8 | 90.4 | 96.4 | 99.0 | 98.6 | 94.8 | 87.0 | <u>95.4</u> |
| Effort | 4 | 100.0 | 98.2 | 83.5 | 98.2 | 100.0 | 99.8 | 99.7 | 73.9 | 94.2 |
| MoeFFD | 4 | 99.6 | 96.7 | 92.2 | 82.0 | 92.7 | 95.4 | 86.4 | 75.8 | 90.1 |
| SimLBR | 4 | 99.1 | 91.0 | 86.8 | 92.2 | 94.6 | 83.4 | 93.9 | 58.9 | 87.5 |
| **Ours** | 4 | 100.0 | 100.0 | 95.8 | 99.2 | 100.0 | 99.7 | 99.7 | 80.6 | **96.9** |

# 4 EXPERIMENTS

## 4.1 DATASETS AND IMPLEMENTATION DETAILS

**Training Set.** For the **generalization evaluation**, we use the official AIGCDetectBenchmark training split Zhong et al. (2023); Yang et al. (2026a). This split contains the full ProGAN Karras et al. (2018) training set consisting of 20 semantic categories. For the **semantic sensitivity evaluation**, we follow the standard protocol proposed by FreqNet Tan et al. (2024). It adopts specific 1-class, 2-class, and 4-class training settings, denoted as (horse), (chair, horse), (car, cat, chair, horse), respectively. Varying the semantic categories used for training allows us to assess how detection performance depends on semantic content. For the **compression robustness evaluation**, we follow prior work Li et al. (2026) and use a 2-class training setting, in which 20% of the training images are compressed with JPEG at a quality factor of 60 while the rest are left uncompressed.

**Test Set.** For the **generalization evaluation**, we use the AIGCDetectBenchmark Zhong et al. (2023) test split, which contains eight GAN-based and eight diffusion-based subsets. The GAN-based subsets include ProGAN Karras et al. (2018), StyleGAN Karras et al. (2019), BigGAN Brock et al. (2019), CycleGAN Zhu et al. (2017), StarGAN Choi et al. (2018), GauGAN Park et al. (2019),

Table 3: Accuracy (%) under **quality-aware robustness evaluation**, with all test images compressed at the same ratio used during training. Baseline results are from CPTFormer Li et al. (2026). Mean ACC is the arithmetic mean over all 17 test subsets.

| Method | InfoMaxGAN | BEGAN | CramerGAN | AttGAN | MMDGAN | RelGAN | S3GAN | SNGGAN | STGGAN |
|---|---|---|---|---|---|---|---|---|---|
| MesoNet | 50.5 | 50.6 | 50.0 | 50.5 | 50.2 | 50.4 | 49.3 | 50.2 | 50.9 |
| FF++ | 74.4 | 30.6 | 75.5 | 64.2 | 76.3 | 61.5 | 54.9 | 71.8 | 82.2 |
| F3Net | 65.9 | 42.6 | 68.9 | 55.9 | 63.7 | 56.3 | 53.1 | 62.1 | 74.9 |
| MAT | 54.5 | 49.8 | 59.7 | 50.1 | 57.8 | 50.8 | 52.8 | 52.8 | 56.7 |
| SBIs | 56.6 | 51.9 | 63.4 | 50.1 | 59.3 | 50.6 | 62.2 | 52.1 | 53.0 |
| ADD | 52.0 | 51.0 | 59.0 | 50.7 | 57.2 | 52.7 | 44.7 | 52.3 | 53.1 |
| QAD | 74.8 | 53.7 | 79.6 | 60.1 | 78.3 | 66.5 | 56.0 | 76.3 | 80.4 |
| FreqNet | 47.7 | 50.0 | 50.5 | 52.6 | 50.5 | 47.8 | 49.1 | 50.6 | 51.7 |
| NPR | 57.2 | 53.4 | 51.8 | 50.9 | 50.7 | 51.8 | 50.4 | 49.7 | 50.8 |
| Fatformer | 64.9 | 53.0 | 64.5 | 50.7 | 61.0 | 51.7 | 72.6 | 55.4 | 57.9 |
| ODDN | 73.1 | 42.3 | 76.1 | 71.2 | 75.9 | 72.5 | 60.5 | 75.5 | 85.0 |
| CPTformer | 85.8 | 73.7 | 87.9 | 58.6 | 88.1 | 61.2 | 75.3 | 79.2 | 80.6 |
| **Ours** | 89.9 | 78.3 | 91.4 | 70.2 | 91.3 | 72.0 | 77.7 | 83.1 | 87.1 |

| Method | ProGAN | StyleGAN | StyleGAN2 | BigGAN | CycleGAN | StarGAN | GauGAN | Deepfake | *Mean*↑ |
|---|---|---|---|---|---|---|---|---|---|
| MesoNet | 51.4 | 51.5 | 54.2 | 52.0 | 53.4 | 50.6 | 53.4 | 50.0 | 51.2 |
| FF++ | 90.4 | 60.4 | 65.2 | 60.5 | 80.0 | 74.4 | 72.3 | 51.0 | 67.3 |
| F3Net | 84.1 | 56.7 | 60.4 | 56.1 | 77.8 | 71.2 | 68.6 | 50.4 | 63.2 |
| MAT | 85.7 | 52.4 | 53.1 | 52.9 | 72.2 | 57.6 | 67.6 | 51.1 | 57.7 |
| SBIs | 88.4 | 51.2 | 52.4 | 55.4 | 74.8 | 53.6 | 78.3 | 51.1 | 59.3 |
| ADD | 70.9 | 48.0 | 48.4 | 51.7 | 72.4 | 55.7 | 64.7 | 51.3 | 55.2 |
| QAD | 86.3 | 55.4 | 57.2 | 59.1 | 77.1 | 79.9 | 65.8 | 55.8 | 69.2 |
| FreqNet | 50.1 | 53.3 | 55.8 | 51.6 | 52.3 | 51.0 | 52.1 | 48.1 | 50.9 |
| NPR | 53.1 | 52.6 | 52.3 | 51.7 | 55.4 | 54.3 | 52.4 | 49.7 | 52.2 |
| Fatformer | 91.3 | 54.8 | 57.7 | 67.0 | 87.3 | 65.1 | 87.1 | 51.8 | 64.3 |
| ODDN | 91.3 | 64.5 | 69.4 | 64.3 | 80.8 | 78.0 | 77.3 | 54.3 | 71.4 |
| CPTformer | 93.2 | 60.8 | 60.7 | 77.7 | 96.9 | 86.9 | 91.3 | 56.7 | 77.4 |
| **Ours** | 93.9 | 64.7 | 61.2 | 73.9 | 95.0 | 94.2 | 90.8 | 61.3 | **80.9** |

StyleGAN2 Karras et al. (2020), and WhichFaceIsReal (WFIR). The diffusion-based subsets include ADM Dhariwal & Nichol (2021), GLIDE, Midjourney, Stable Diffusion v1.4 Rombach et al. (2022) and v1.5, VQDM Gu et al. (2022), Wukong, and DALL-E 2 Ramesh et al. (2022). For the **semantic sensitivity evaluation** and the **compression robustness evaluation**, we use the test sets from FreqNet Tan et al. (2024) and CPTFormer Li et al. (2026), respectively.

**Implementation Details.** Following prior work Yang et al. (2026a); Dhakal et al. (2026); Liu et al. (2024); Yan et al. (2025b), we use a pretrained CLIP ViT-L/14 Radford et al. (2021) as the backbone. The backbone is evenly partitioned into three stages, each consisting of 8 ViT blocks, and each stage is equipped with a Bidirectional Chroma-RGB Interaction module. During training, we apply random $224 \times 224$ crops and random horizontal flips. At test time, we use a $224 \times 224$ center crop. We train the detector for 10 epochs using AdamW Kingma & Ba (2015), with a learning rate of $1 \times 10^{-4}$ and a batch size of 96. We report accuracy (ACC) and the reliability score (Rel.), following prior work Dhakal et al. (2026); Yang et al. (2026a); Tan et al. (2024).

### 4.2 Performance Evaluation

**Generalization Evaluation.** Table 1 reports cross-generator generalization results for the 16 AIGCDetectBenchmark subsets. Mean ACC and reliability are computed jointly across all subsets. ChromaGuard achieves the highest mean ACC of 95.91%. It outperforms HSIC Yang et al. (2026a), which suppresses semantic information through feature-level constraints, by 3.50 percentage points. ChromaGuard also achieves the highest reliability score of 8.38, an improvement of 2.22 over SimLBR. Together, these improvements support the effectiveness of image-level chroma guidance for cross-generator generalization.

Table 4: Accuracy (%) under **quality-agnostic robustness evaluation**, with each test image compressed at a randomly selected ratio. Baseline results are from CPTFormer Li et al. (2026).

| Method | InfoMaxGAN | BEGAN | CramerGAN | AttGAN | MMDGAN | RelGAN | S3GAN | SNGGAN | STGGAN |
|---|---|---|---|---|---|---|---|---|---|
| MesoNet | 46.3 | 44.3 | 59.7 | 60.0 | 59.8 | 58.7 | 47.8 | 56.3 | 69.1 |
| FF++ | 66.9 | 37.7 | 79.4 | 56.6 | 77.1 | 60.5 | 55.0 | 69.2 | 79.8 |
| F3Net | 58.0 | 48.8 | 61.9 | 51.5 | 59.3 | 53.2 | 52.0 | 54.9 | 61.1 |
| MAT | 54.2 | 49.9 | 59.6 | 50.5 | 57.6 | 51.2 | 52.1 | 52.7 | 57.8 |
| SBI | 56.6 | 51.9 | 63.4 | 50.1 | 59.3 | 50.6 | 62.2 | 52.1 | 53.0 |
| ADD | 51.8 | 50.9 | 59.0 | 50.7 | 57.1 | 52.8 | 45.0 | 52.3 | 52.9 |
| QAD | 72.3 | 55.2 | 80.0 | 61.5 | 78.3 | 65.5 | 54.5 | 76.5 | 79.2 |
| FreqNet | 49.4 | 59.5 | 48.8 | 51.6 | 49.3 | 48.5 | 47.1 | 50.6 | 50.6 |
| NPR | 58.7 | 54.8 | 52.1 | 53.3 | 51.8 | 53.3 | 50.8 | 50.2 | 51.3 |
| Fatformer | 65.6 | 53.4 | 65.2 | 50.9 | 62.1 | 51.8 | 74.5 | 56.0 | 58.1 |
| ODDN | 72.1 | 44.1 | 76.8 | 68.1 | 76.5 | 73.3 | 58.0 | 75.6 | 83.5 |
| CPTformer | 85.8 | 71.9 | 87.6 | 57.8 | 87.3 | 60.8 | 73.9 | 79.4 | 79.9 |
| **Ours** | 88.5 | 74.5 | 90.0 | 67.3 | 88.9 | 70.0 | 74.5 | 80.6 | 84.4 |

| Method | ProGAN | StyleGAN | StyleGAN2 | BigGAN | CycleGAN | StarGAN | GauGAN | Deepfake | *Mean*↑ |
|---|---|---|---|---|---|---|---|---|---|
| MesoNet | 55.0 | 51.0 | 49.9 | 53.9 | 60.5 | 64.8 | 49.9 | 51.1 | 53.3 |
| FF++ | 87.8 | 55.1 | 59.8 | 57.1 | 79.9 | 75.6 | 71.6 | 52.0 | 66.0 |
| F3Net | 83.4 | 52.7 | 54.9 | 55.0 | 73.7 | 65.9 | 66.7 | 52.4 | 59.3 |
| MAT | 86.1 | 52.3 | 53.0 | 52.7 | 70.3 | 58.2 | 68.0 | 51.3 | 57.7 |
| SBI | 88.6 | 51.3 | 52.4 | 55.7 | 76.0 | 53.9 | 78.1 | 51.2 | 59.4 |
| ADD | 70.2 | 48.0 | 48.7 | 51.8 | 71.9 | 55.5 | 65.1 | 51.3 | 57.9 |
| QAD | 86.4 | 56.4 | 58.0 | 57.4 | 82.6 | 77.8 | 63.5 | 56.5 | 68.3 |
| FreqNet | 50.9 | 55.9 | 62.3 | 53.4 | 51.2 | 50.0 | 53.4 | 48.7 | 51.8 |
| NPR | 53.5 | 53.8 | 54.0 | 53.4 | 50.8 | 54.7 | 52.7 | 49.7 | 52.9 |
| Fatformer | 92.0 | 55.3 | 58.0 | 67.7 | 90.0 | 65.9 | 87.5 | 52.0 | 65.1 |
| ODDN | 90.8 | 61.1 | 65.9 | 63.9 | 83.5 | 77.0 | 72.9 | 55.0 | 70.7 |
| CPTformer | 92.4 | 59.1 | 58.5 | 75.9 | 95.8 | 85.8 | 90.7 | 55.5 | <u>76.3</u> |
| **Ours** | 93.2 | 62.5 | 59.5 | 70.5 | 94.2 | 93.7 | 90.0 | 60.6 | **79.0** |

**Semantic Sensitivity Evaluation.** Table 2 reports accuracy under the three training settings. This comparison assesses how detection performance varies with the semantic categories represented in the training data. ChromaGuard achieves mean ACC values of 96.9%, 96.7%, and 96.9% in the 1-class, 2-class, and 4-class settings, respectively. These values exceed those of Rine, the strongest baseline in each setting, by 2.3, 2.0, and 1.5 percentage points, respectively. Although accuracy varies across individual test subsets, the mean accuracy remains stable across the three training settings. This overall stability suggests that ChromaGuard is robust to changes in the semantic diversity of the training data under the evaluated settings.

**Compression Robustness Evaluation.** Tables 3 and 4 report compression robustness under quality-aware and quality-agnostic settings, respectively. The quality-aware setting uses the compression ratio applied during training. The quality-agnostic setting uses a randomly selected compression ratio for test images. ChromaGuard achieves the highest mean ACC in both settings, reaching 80.9% and 79.0%, respectively. These values exceed those of CPTFormer Li et al. (2026), the strongest baseline, by 3.5% and 2.7%, respectively. Mean ACC is 1.9 percentage points lower in the quality-agnostic setting than in the quality-aware setting. ChromaGuard retains an advantage over the baselines under both compression settings, demonstrating its robustness to image compression.

**Ablation Studies** We assess the contribution of each component on AIGCDetectBenchmark. The baseline model ($C_b$) achieves a mean ACC of 85.03% (Table 5). Adding the bidirectional Chroma-RGB interaction module ($C_b + C_i$) increases mean ACC to 93.96%. Adding only the DCT high-frequency adapter to the baseline ($C_b + C_d$) raises mean ACC to 93.41%. The full model ($C_b + C_i + C_d$) achieves the highest mean ACC of 95.91%, exceeding the baseline by 10.88%. It also exceeds the interaction-only and adapter-only variants by 1.95% and 2.50%, respectively. These results support the complementary contributions of the two components to overall generalization, with gains that vary across generators.

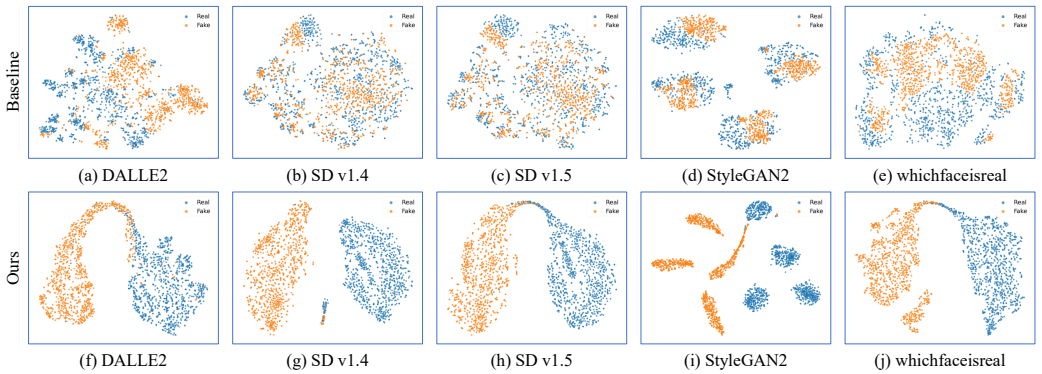
Table 5: Effectiveness of different components. $C_b$: baseline; $C_i$: Chroma-RGB guidance; $C_d$: DCT high-frequency adapter. The last column of Panel B reports the overall mean computed jointly across all 16 generators. An additional ablation study of image components is provided in the **Appendix**.

**Panel A: GAN-based Generators**

| $C_b$ | $C_i$ | $C_d$ | ProGAN | StyleGAN | BigGAN | CycleGAN | StarGAN | GauGAN | StyleGAN2 | WFIR |
|---|---|---|---|---|---|---|---|---|---|---|
| ✓ | | | 99.96 | 88.70 | 95.53 | 97.65 | 98.27 | 95.89 | 83.12 | 88.55 |
| ✓ | ✓ | | 100.0 | 99.29 | 97.92 | 99.05 | 100.0 | 95.22 | 97.63 | 98.70 |
| ✓ | | ✓ | 100.0 | 92.81 | 99.02 | 99.17 | 100.0 | 98.96 | 97.96 | 97.90 |
| ✓ | ✓ | ✓ | 100.0 | 98.89 | 93.78 | 98.18 | 99.80 | 92.08 | 98.98 | 98.75 |

**Panel B: DM-based Generators**

| $C_b$ | $C_i$ | $C_d$ | ADM | Glide | Midjourney | SDv1.4 | SDv1.5 | VQDM | Wukong | DALLE2 | *Mean*↑ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ✓ | | | 80.08 | 89.34 | 56.83 | 76.78 | 76.81 | 89.31 | 83.12 | 60.55 | 85.03 |
| ✓ | ✓ | | 89.29 | 94.14 | 58.17 | 97.16 | 96.79 | 97.32 | 96.55 | 86.05 | 93.96 |
| ✓ | | ✓ | 84.40 | 95.93 | 59.88 | 97.28 | 97.08 | 93.53 | 95.96 | 84.65 | 93.41 |
| ✓ | ✓ | ✓ | 92.88 | 95.49 | 77.93 | 98.92 | 98.59 | 98.06 | 98.78 | 93.40 | **95.91** |




(a) DALLE2 (b) SD v1.4 (c) SD v1.5 (d) StyleGAN2 (e) whichfaceisreal

(f) DALLE2 (g) SD v1.4 (h) SD v1.5 (i) StyleGAN2 (j) whichfaceisreal

Figure 3: T-SNE visualizations of feature distributions. Blue and orange dots denote real and synthetic images, respectively. Compared with the baseline, ChromaGuard produces better-separated feature clusters with minimal overlap between real and synthetic images.

### 4.3 Visualization

We use t-SNE van der Maaten & Hinton (2008) to visualize real and synthetic image features under cross-generator shifts. The five test subsets contain images from generators not seen during training. Figure 3 shows substantial overlap between real and synthetic samples in the baseline feature projections. In contrast, ChromaGuard produces substantially better-separated feature clusters, with real and synthetic samples forming distinct groups and only minimal overlap. This clear separation is consistently observed across all five unseen datasets, providing qualitative evidence that ChromaGuard learns more transferable forensic representations.

## 5 Conclusion

We address semantic reliance in synthetic image detection from an image-level perspective. Based on our observation that YCbCr chroma is only weakly associated with semantic content, we propose ChromaGuard. From the outset of feature extraction, a simple bidirectional Chroma-RGB interaction module progressively guides the detector toward cues weakly associated with semantic content. A DCT high-frequency adapter in the RGB branch further enhances sensitivity to synthesis artifacts. Evaluations of cross-generator generalization, semantic sensitivity, and robustness demonstrate strong performance and reliability, supporting the effectiveness of chroma guidance.

### AI USE STATEMENT

We used generative AI tools for paper polishing to improve the grammar, clarity, and readability of the manuscript. The authors reviewed all AI-assisted revisions to ensure that they preserved the intended scientific meaning and introduced no unsupported claims. We take full responsibility for the final content of this work, including all text, claims, and artifacts produced with the assistance of generative AI.

### REPRODUCIBILITY STATEMENT

To support reproducibility, Section 3 describes detailed model architecture, the bidirectional Chroma-RGB interaction module, the DCT high-frequency adapter, and the training objective. Section 4 provides the dataset sources, training and test settings, image preprocessing procedures, and optimization hyperparameters.

## REFERENCES

Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale GAN training for high fidelity natural image synthesis. In *ICLR*, 2019.

Yunjey Choi, Min-Je Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *CVPR*, pp. 8789–8797, 2018.

Aayush Dhakal, Subash Khanal, Srikumar Sastry, Jacob Arndt, Philipe Ambrozio Dias, Dalton Lunga, and Nathan Jacobs. Simlbr: Learning to detect fake images by learning to detect real images. In *CVPR*, 2026.

Prafulla Dhariwal and Alexander Quinn Nichol. Diffusion models beat gans on image synthesis. In *NeurIPS*, pp. 8780–8794, 2021.

Joel Frank, Thorsten Eisenhofer, Lea Schönherr, Asja Fischer, Dorothea Kolossa, and Thorsten Holz. Leveraging frequency analysis for deep fake image recognition. In *ICML*, volume 119, pp. 3247–3258, 2020.

Xinghe Fu, Zhiyuan Yan, Taiping Yao, Shen Chen, and Xi Li. Exploring unbiased deepfake detection via token-level shuffling and mixing. In *AAAI*, 2025.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.

Shuyang Gu, Dong Chen, Jianmin Bao, Fang Wen, Bo Zhang, Dongdong Chen, Lu Yuan, and Baining Guo. Vector quantized diffusion model for text-to-image synthesis. In *CVPR*, pp. 10696–10706, 2022.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, volume 33, pp. 6840–6851, 2020.

Yonghyun Jeong, Doyeon Kim, Seungjai Min, Seongho Joe, Youngjune Gwon, and Jongwon Choi. Bihpf: Bilateral high-pass filters for robust deepfake detection. In *WACV*, pp. 2878–2887, 2022.

Peiqi Jiang, Hongtao Xie, Lingyun Yu, Guoqing Jin, and Yongdong Zhang. Exploring bi-level inconsistency via blended images for generalizable face forgery detection. *TIFS*, 19:6573–6588, 2024.

Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. In *ICLR*, 2018.

Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *CVPR*, pp. 4396–4405, 2019.

Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *CVPR*, pp. 8107–8116, 2020.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.

Christos Koutlis and Symeon Papadopoulos. Leveraging representations from intermediate encoder-blocks for synthetic image detection. In *ECCV*, volume 15130, pp. 394–411, 2024.

Jiaming Li, Hongtao Xie, Jiahong Li, Zhongyuan Wang, and Yongdong Zhang. Frequency-aware discriminative feature learning supervised by single-center loss for face forgery detection. In *CVPR*, pp. 6458–6467, 2021.

Kai Li, Wenqi Ren, Wei Wang, and Xiaochun cao. Detecting compressed ai-generated images via phase spectrum robustness. In *CVPR*, 2026.

Lingzhi Li, Jianmin Bao, Ting Zhang, Hao Yang, Dong Chen, Fang Wen, and Baining Guo. Face x-ray for more general face forgery detection. In *CVPR*, pp. 5000–5009, 2020.

Bo Liu, Qiao Qin, and Qinghui He. Causalclip: Causally-informed feature disentanglement and filtering for generalizable detection of generated images. In *AAAI*, pp. 7069–7077, 2026.

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.

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

Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *ICML*, volume 139, pp. 8162–8171, 2021.

Utkarsh Ojha, Yuheng Li, and Yong Jae Lee. Towards universal fake image detectors that generalize across generative models. In *CVPR*, pp. 24480–24489, 2023.

Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization. In *CVPR*, pp. 2337–2346, 2019.

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, 2020.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, volume 139, pp. 8748–8763, 2021.

Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *ArXiv*, 2022.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, pp. 10674–10685, 2022.

Kaede Shiohara and Toshihiko Yamasaki. Detecting deepfakes with self-blended images. In *CVPR*, pp. 18720–18729, 2022.

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 *AAAI*, pp. 5052–5060, 2024.

Renshuai Tao, Chuangchuang Tan, Huan Liu, Jiakai Wang, Haotong Qin, Yakun Chang, Wei Wang, Rongrong Ni, and Yao Zhao. Sagnet: Decoupling semantic-agnostic artifacts from limited training data for robust generalization in deepfake detection. *TIFS*, 20:6429–6442, 2025.

Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *JMLR*, 9(86):2579–2605, 2008.

Jun Wang, Zitong Yu, Chaomeng Chen, Lingchen Gu, Wenbo Wan, Jiantao Zhou, and Weiming Zhang. Quality-guided forgery adapter for generalizable aigc image detection. *TIFS*, 21, 2026.

Sheng-Yu Wang, Oliver Wang, Richard Zhang, Andrew Owens, and Alexei A. Efros. Cnn-generated images are surprisingly easy to spot... for now. In *CVPR*, pp. 8692–8701, 2020.

Shilin Yan, Ouxiang Li, Jiayin Cai, Yanbin Hao, Xiaolong Jiang, Yao Hu, and Weidi Xie. A sanity check for ai-generated image detection. In *ICLR*, volume 2025, pp. 70702–70720, 2025a.

Zhiyuan Yan, Jiangming Wang, Peng Jin, Ke-Yue Zhang, Chengchun Liu, Shen Chen, Taiping Yao, Shouhong Ding, Baoyuan Wu, and Li Yuan. Orthogonal subspace decomposition for generalizable ai-generated image detection. In *ICML*, 2025b.

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

Yawen Yang, Feng Li, Shuqi Kong, Yunfeng Diao, Xinjian Gao, Zenglin Shi, and Meng Wang. Layer consistency matters: Elegant latent transition discrepancy for generalizable synthetic image detection. In *CVPR*, 2026b.

Dengyong Zhang, Miao Hu, Jiaxin Chen, Changsheng Chen, Jin Wang, Yun Song, Gaobo Yang, Xin Liao, and Xiangling Ding. Falcon-net: Feature aggregation of local patterns for ai-generated image detection. *TIFS*, 21:3842–3855, 2026.

Haifeng Zhang, Qinghui He, Xiuli Bi, Weisheng Li, Bo Liu, and Bin Xiao. Towards universal ai-generated image detection by variational information bottleneck network. In *CVPR*, pp. 23828–23837, 2025.

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

Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *ICCV*, pp. 2242–2251, 2017.

# A Appendix

**Ablation Studies.** To assess how the choice of image component affects cross-generator generalization, we compare Chroma with DWT and DCT in Appendix Table 6. Chroma achieves the highest overall mean accuracy and reliability, reaching 95.91% and 8.38, respectively, compared with DWT (93.48%, 4.06) and DCT (92.99%, 4.31). On the eight diffusion-based generators, Chroma outperforms DWT on every subset. On the GAN-based generators, DWT is stronger on BigGAN, CycleGAN, StarGAN, GauGAN, and WFIR, whereas Chroma performs better on StyleGAN and StyleGAN2 and matches DWT on ProGAN. These results indicate that Chroma provides a strong image-level guidance signal for cross-generator detection.

**Semantic Sensitivity Evaluation.** Table 7 complements the accuracy results in Table 2 by reporting average precision (AP) under the same semantic sensitivity evaluation. We evaluate each detector after training on one, two, or four semantic categories ($\#n \in 1, 2, 4$) and test it on eight GAN-based subsets. AP summarizes the detector's ranking quality across decision thresholds, providing an additional assessment of whether its performance remains stable as the semantic diversity of the training data changes.

ChromaGuard achieves mean AP values of 99.6%, 99.5%, and 99.7% in the 1-class, 2-class, and 4-class settings, respectively, remaining within 0.1–0.3 percentage points of the best or second-best result in each setting. Moreover, it obtains an AP of at least 99.2% on seven of the eight test subsets across all three settings, while maintaining 97.0–97.6% AP on the more challenging Deepfake subset. The narrow 0.2-point variation in mean AP across the three training settings further supports the low sensitivity of ChromaGuard to the semantic content.

Table 6: Image-component ablation results for cross-generator generalization. Given the available computational resources, we compare Chroma with DCT, which has the second-weakest semantic association among all components in Table 1, and DWT, another frequency-domain transform.

**Panel A: GAN-based Generators**

| Method | Type | ProGAN | StyleGAN | BigGAN | CycleGAN | StarGAN | GauGAN | StyleGAN2 | WFIR |
|---|---|---|---|---|---|---|---|---|---|
| DWT | Freq | 100.00 | 95.59 | 98.30 | 99.43 | 99.92 | 98.63 | 95.92 | 98.80 |
| DCT | Freq | 100.00 | 93.31 | 98.45 | 99.43 | 99.95 | 98.39 | 98.02 | 98.05 |
| **Chroma** | Spat | 100.0 | 98.89 | 93.78 | 98.18 | 99.80 | 92.08 | 98.98 | 98.75 |

**Panel B: DM-based Generators**

| Method | ADM | Glide | Midjourney | SDv1.4 | SDv1.5 | VQDM | Wukong | DALLE2 | *Mean*↑ | *Rel.*↑ |
|---|---|---|---|---|---|---|---|---|---|---|
| DWT | 89.58 | 94.94 | 61.21 | 97.77 | 97.73 | 96.98 | 97.18 | 73.75 | 93.48 | 4.06 |
| DCT | 83.94 | 91.56 | 61.79 | 96.93 | 96.89 | 95.85 | 94.40 | 80.85 | 92.99 | 4.31 |
| **Chroma** | 92.88 | 95.49 | 77.93 | 98.92 | 98.59 | 98.06 | 98.78 | 93.40 | **95.91** | **8.38** |

Table 7: Average precision (AP) comparison for semantic sensitivity evaluation.

| Methods | #n | ProGAN | StyleGAN | StyleGAN2 | BigGAN | CycleGAN | StarGAN | GauGAN | Deepfake | *Mean*↑ |
|---|---|---|---|---|---|---|---|---|---|---|
| CNNSpot | 1 | 99.7 | 97.8 | 96.4 | 70.8 | 58.2 | 99.9 | 64.2 | 91.8 | 84.9 |
| F3Net | 1 | 100.0 | 99.7 | 99.2 | 60.4 | 54.5 | 100.0 | 46.5 | 93.6 | 81.7 |
| SelfBlend | 1 | 95.1 | 65.2 | 65.3 | 66.1 | 88.4 | 80.4 | 91.1 | 72.4 | 78.0 |
| UnivFD | 1 | 100.0 | 98.2 | 94.0 | 97.9 | 99.8 | 99.4 | 99.8 | 88.7 | 97.2 |
| FatFormer | 1 | 100.0 | 99.7 | 99.2 | 99.0 | 100.0 | 99.6 | 100.0 | 92.4 | 98.7 |
| FreqNet | 1 | 99.9 | 98.7 | 97.9 | 93.1 | 99.1 | 98.4 | 99.6 | 94.4 | 97.6 |
| NPR | 1 | 99.8 | 99.6 | 97.3 | 83.3 | 93.6 | 100.0 | 59.3 | 90.9 | 90.5 |
| Rine | 1 | 100.0 | 100.0 | 97.9 | 97.7 | 100.0 | 100.0 | 99.8 | 98.5 | <u>99.2</u> |
| Effort | 1 | 99.8 | 94.7 | 87.8 | 99.0 | 100.0 | 100.0 | 99.9 | 95.0 | 97.0 |
| MoeFFD | 1 | 99.9 | 98.8 | 95.8 | 88.4 | 99.7 | 99.0 | 93.9 | 89.6 | 95.6 |
| SimLBR | 1 | 99.8 | 88.7 | 89.9 | 98.9 | 99.8 | 91.7 | 100.0 | 81.6 | 93.8 |
| **Ours** | 1 | 100.0 | 100.0 | 99.6 | 99.7 | 100.0 | 100.0 | 100.0 | 97.4 | **99.6** |
| CNNSpot | 2 | 100.0 | 98.4 | 93.1 | 78.9 | 73.9 | 100.0 | 80.3 | 91.3 | 89.5 |
| F3Net | 2 | 99.8 | 99.0 | 98.6 | 63.5 | 63.5 | 100.0 | 49.2 | 85.9 | 82.1 |
| SelfBlend | 2 | 95.6 | 64.5 | 66.7 | 65.8 | 90.9 | 82.4 | 93.2 | 72.8 | 79.0 |
| UnivFD | 2 | 100.0 | 98.4 | 97.5 | 98.1 | 99.8 | 99.6 | 99.9 | 90.0 | 97.9 |
| FatFormer | 2 | 100.0 | 99.9 | 99.7 | 99.8 | 100.0 | 100.0 | 100.0 | 98.6 | **99.8** |
| FreqNet | 2 | 100.0 | 98.9 | 98.1 | 96.0 | 99.8 | 100.0 | 98.8 | 92.0 | 97.9 |
| NPR | 2 | 99.8 | 99.6 | 98.1 | 84.0 | 96.9 | 100.0 | 60.4 | 94.9 | 91.7 |
| Rine | 2 | 100.0 | 100.0 | 99.9 | 99.8 | 100.0 | 100.0 | 99.9 | 98.5 | <u>99.8</u> |
| Effort | 2 | 100.0 | 99.3 | 96.6 | 100.0 | 100.0 | 100.0 | 100.0 | 97.4 | 99.2 |
| MoeFFD | 2 | 100.0 | 99.2 | 98.6 | 94.5 | 99.9 | 99.9 | 97.2 | 87.9 | 97.2 |
| SimLBR | 2 | 100.0 | 91.3 | 90.8 | 99.3 | 99.5 | 93.2 | 100.0 | 80.1 | 94.3 |
| **Ours** | 2 | 100.0 | 100.0 | 99.2 | 99.9 | 100.0 | 100.0 | 100.0 | 97.0 | 99.5 |
| CNNSpot | 4 | 100.0 | 99.5 | 96.7 | 76.2 | 68.8 | 100.0 | 73.8 | 95.8 | 88.9 |
| F3Net | 4 | 100.0 | 99.4 | 99.3 | 76.3 | 77.4 | 100.0 | 56.7 | 94.8 | 88.0 |
| SelfBlend | 4 | 99.7 | 86.4 | 80.1 | 76.5 | 93.7 | 90.1 | 94.6 | 81.9 | 87.9 |
| UnivFD | 4 | 100.0 | 99.9 | 99.1 | 98.4 | 99.7 | 99.7 | 100.0 | 92.5 | 98.6 |
| FatFormer | 4 | 100.0 | 100.0 | 99.9 | 99.9 | 100.0 | 100.0 | 100.0 | 96.0 | 99.5 |
| FreqNet | 4 | 100.0 | 99.7 | 99.5 | 96.0 | 99.6 | 99.8 | 98.6 | 94.4 | 98.5 |
| NPR | 4 | 100.0 | 99.8 | 99.6 | 87.4 | 96.3 | 100.0 | 66.6 | 96.9 | 93.3 |
| Rine | 4 | 100.0 | 100.0 | 99.9 | 99.8 | 100.0 | 100.0 | 99.9 | 98.4 | **99.8** |
| Effort | 4 | 100.0 | 100.0 | 98.4 | 100.0 | 100.0 | 100.0 | 100.0 | 97.7 | 99.5 |
| MoeFFD | 4 | 100.0 | 99.6 | 99.1 | 89.9 | 99.6 | 100.0 | 93.3 | 88.0 | 96.2 |
| SimLBR | 4 | 100.0 | 98.8 | 97.6 | 99.3 | 99.3 | 94.0 | 100.0 | 80.1 | 96.1 |
| **Ours** | 4 | 100.0 | 100.0 | 99.8 | 100.0 | 100.0 | 100.0 | 100.0 | 97.6 | <u>99.7</u> |