PHYSCONS: PHYSICS-AWARE CONSISTENCY MODELING FOR ROBUST AI-GENERATED IMAGE DETECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Detecting AI-generated images from unseen generators remains challenging, as existing methods often overfit to generator-specific artifacts that evolve rapidly. We observe that while generative models change over time, the physical laws governing real-world image formation—such as optics, sensor noise, and illumination—remain invariant. Based on this observation, we propose PhysCons, a physics-aware consistency modeling framework that detects synthetic images by identifying violations of intrinsic physical consistency, rather than modeling generator-dependent signatures. PhysCons jointly exploits three coupled physical dimensions—spectral–spatial structure, luminance–chrominance coupling, and noise–texture statistics—to expose the physical decoupling inherent in generative processes. Extensive experiments on GenImage and AIGCDetectBenchmark demonstrate state-of-the-art generalization of PhysCons to unseen generators. Notably, PhysCons achieves this performance with only 0.05M parameters and 722 FPS, offering an exceptional balance between robustness, accuracy, and efficiency.

1 INTRODUCTION

The rapid advancement of AI-generated image technologies has reshaped online content creation and dissemination. Recent generative models (Goodfellow et al., 2020; Ho et al., 2020; Rombach et al., 2022; Black Forest Labs, 2024; Yan et al., 2025c) can produce highly realistic images that are indistinguishable from real photographs, raising serious risks such as identity impersonation, visual misinformation, financial scams, and reputational abuse. Consequently, reliable and robust detection frameworks are essential for safeguarding visual authenticity in real-world applications.

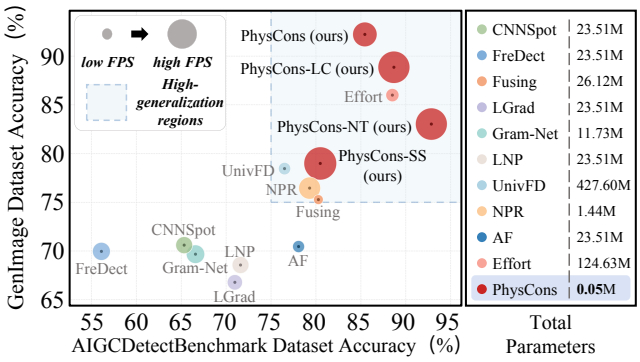


Figure 1: Illustration of performance vs. efficiency trade-off on AIGCDetect (x-axis) and GenImage (y-axis) benchmarks. Bubble size indicates inference speed (FPS). PhysCons achieves superior cross-generator generalization with only 0.05M parameters, orders of magnitude smaller than competitors, offering a favorable accuracy–efficiency trade-off.

The research community has proposed many detection methods, ranging from mining low-level artifacts in frequency or noise domains (Rombach et al., 2022; Frank et al., 2020; Liu et al., 2022) to leveraging pre-trained priors from diffusion models or CLIP (Radford et al., 2021; Wang et al., 2023; Ojha et al., 2023; Yan et al., 2025a). However, most existing detectors rely on modeling statistical discrepancies in the training distribution, making them data-dependent and prone to overfitting generator-specific artifacts. As a result, their performance degrades sharply on unseen models as generation techniques evolve, a limitation highlighted by recent benchmarks such as Chameleon (Yan et al., 2025a).

Motivated by these challenges, we argue that effective detection should be grounded in invariant physical principles underlying real-world image formation, instead of chasing evolving generator-

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.

specific artifacts. Unlike conventional descriptors (e.g., LBP or GLCM) that capture generic statistical patterns, we focus on cross-signal consistency induced by the image formation process, as it is governed by physically grounded coupling relationships that remain stable across imaging conditions. To this end, we propose PhysCons, a physics-aware consistency modeling framework for robust detection of state-of-the-art AI-generated images. Our key insight is that real images arise from causal physical interactions (e.g., optics and lighting), inducing consistent relationships across signals and feature extraction pathways. In contrast, deep generative models learn statistical associations without explicit physical constraints, often capturing superficial correlations rather than the underlying causal generative process (Schölkopf et al., 2021). Consequently, AI-generated images fail to preserve such cross-signal consistency. Specifically, we leverage three physical cues:

- **The spectral–spatial edge structural consistency.** In natural images, salient structures tend to arise at spatial locations where the phases of multiple frequency components are highly aligned (Kovesi, 1999), and these locations often coincide with peaks in spatial gradient magnitude, in line with the local energy model (Morrone & Owens, 1987).
- **The luminance-chrominance coupling.** Although natural image statistics suggest that Luminance (L) and Chrominance (C) edge magnitudes are statistically independent (i.e., the intensity of one cannot linearly predict the other) (Hansen & Gegenfurtner, 2009), they share common physical origins, such as object boundaries and surface textures. Consequently, L and C signals exhibit strong spatial structural consistency in real images (Fine et al., 2003; Hansen & Gegenfurtner, 2009).
- **The noise-texture statistical dependency.** In real images, signal-dependent shot noise and ISP denoising induce a strong dependence between noise residuals and local texture complexity (Liu et al., 2014).

Leveraging the above physical cues, our framework casts AI-generated image detection as a physics-aware consistency modeling problem, where synthetic images are identified by violations of intrinsic physical constraints underlying real-world image formation. Specifically, PhysCons projects physically related feature pairs—covering spectral-spatial structure, luminance–chrominance relations, and noise–texture statistics—into three shared latent spaces and explicitly models their cross-signal consistency deviations. The framework is optimized with a joint objective that combines a standard classification loss $\mathcal{L}_{\text{cls}}$ and a dedicated violation loss $\mathcal{L}_{\text{vio}}$. The violation loss enforces physical consistency in real images while encouraging generated images to exhibit pronounced violations. By anchoring detection to intrinsic physical principles of image formation rather than generator-specific artifacts, PhysCons achieves superior generalization to unseen generative models, while remaining lightweight and highly efficient (Fig. 1), and naturally provides interpretable, pixel-level evidence of physical violations (see Appendix D.2). Overall, our approach can be summarized as:

- **A physics-grounded detection paradigm.** We propose a new paradigm that detects AI-generated images via physical constraint violations, rather than relying on generator-specific artifacts.
- **A unified physics-aware framework.** We introduce PhysCons, which captures consistency across spectral–spatial, luminance–chrominance, and noise–texture via explicit violation modeling.
- **Strong generalization, efficiency, and interpretability.** Experiments show state-of-the-art generalization to unseen generators, high-throughput inference, and intuitive physical violation maps.

2 Related Work

Existing AI-generated image detection approaches can be taxonomized into three categories.

Low-level Artifact Mining. Early works target spatial and frequency traces, e.g., CNNSpot (Wang et al., 2020) and FreDect (Frank et al., 2020). Later studies explore finer features: LGrad (Tan et al., 2023) and NPR (Tan et al., 2024) use gradient and pixel statistics, while Gram-Net (Liu et al., 2020) and LNP (Liu et al., 2022) target global texture and noise. Fusing (Ju et al., 2022) integrates global–local cues. However, these methods rely on artifacts less pronounced in advanced models.

Leveraging Pre-trained Priors. Recent approaches leverage pre-trained representations. One stream exploits reconstruction discrepancies in diffusion models (Wang et al., 2023; Luo et al., 2024), positing that real images are harder to reconstruct. Another stream utilizes vision-language models

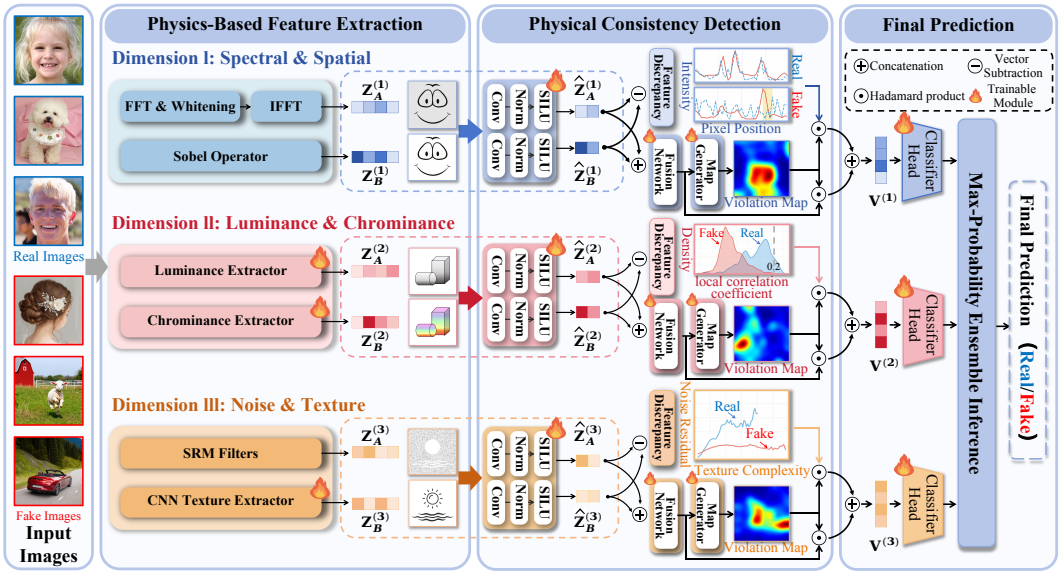


Figure 2: Overview of the PhysCons feed-forward pipeline. The process includes three key stages: (1) extracting three types of feature pairs using $\phi_A^{(k)}$ and $\phi_B^{(k)}$ (Table 1); (2) projecting each feature pair into a shared latent space and modeling cross-signal physical-consistency violations via a learned fusion network to generate spatial physical violation maps; and (3) performing branch-wise classification and aggregating predictions via max-ensemble rule for the final real/fake decision.

like CLIP. UnivFD (Ojha et al., 2023) tunes a linear head on frozen encoders, while AIDE (Yan et al., 2025a) and Effort (Yan et al., 2025b) integrate pixel-level features or orthogonal decomposition to capture forgery traces within semantic embeddings. Despite improvements, these methods model statistical deviations, which remain brittle to domain shifts.

Dataset Alignment. To mitigate data bias, recent works align real and synthetic distributions. FakeInversion (Cazenavette et al., 2024) and DRCT (Chen et al., 2024) generate aligned data via inversion/reconstruction to eliminate non-semantic differences. Aligned-Forensics (AF) (Sundara Rajan et al., 2025) employs VAEs to construct clean training sets. While reducing bias, these approaches do not explicitly enforce learning intrinsic forgery indicators.

3 The Proposed PhysCons Approach

3.1 Overview

While real images are formed through a unified image formation pipeline, AI-generated images are synthesized by data-driven generative models that do not explicitly enforce such cross-signal couplings during training, often resulting in subtle but detectable inconsistencies across different representations. Therefore, we detect AI-generated images by identifying violations of these cross-signal consistencies. To this end, we consider K pairs of physically related feature extractors, $\Phi = \{(\phi_A^{(k)}, \phi_B^{(k)})\}_{k=1}^K$, where each pair is designed to encode the same underlying physical property from different perspectives (e.g., spectral-spatial structure, luminance-chrominance coupling, and noise–texture statistics). For real images, the outputs of each extractor pair are expected to be cross-signal consistent, whereas such consistency is frequently disrupted in AI-generated content. Based on this observation, we introduce a set of *physical consistency discriminators* $\{\mathcal{D}_k\}_{k=1}^K$. Each discriminator $\mathcal{D}_k$ assesses the degree of physical inconsistency between the two features in the pair and maps it to a score that reflects the likelihood of the image $\mathbf{x}$ being fake:

$$\mathcal{P}_k(y = 1 \mid \mathbf{x}) = \mathcal{D}_k\left(\phi_A^{(k)}(\mathbf{x}),\, \phi_B^{(k)}(\mathbf{x})\right). \tag{1}$$

Importantly, AI-generated images often violate only a subset of physical constraints, causing single-consistency detectors to miss violations that would otherwise indicate the image as fake. To address

this, we adopt a max-ensemble rule that flags an image $\mathbf{x}$ as fake if *any* inconsistency is detected:

$$\mathcal{P}_{\text{fake}}(\mathbf{x}) = \max_{k=1}^{K} \mathcal{P}_k(y = 1 \mid \mathbf{x}), \tag{2}$$

where $\mathcal{P}_k(y = 1 \mid \mathbf{x})$ denotes the fake probability from the k-th physical constraint, and the max operation implements a logical OR over all constraints. As shown in Fig. 2, our framework decouples features along multiple physical dimensions and aggregates their consistency violations for detection.

3.2 Physics-Aware Feature Consistency Modeling

Table 1: Paired feature extractors designed to capture three types of physical cues: spectral–spatial edge structure, luminance–chrominance coupling, and noise–texture statistics. Here, $\mathcal{G}(\cdot)$ denotes grayscale conversion, $\mathcal{F}(\cdot)$ and $\mathcal{F}^{-1}(\cdot)$ denote the Fourier transform and its inverse, $\mathrm{Re}(\cdot)$ denotes the real part of a complex-valued signal, $\mathbf{S}_x$ and $\mathbf{S}_y$ are Sobel filters for horizontal and vertical gradients, $*$ denotes convolution, $\mathbf{x}_R$, $\mathbf{x}_G$, and $\mathbf{x}_B$ denote the RGB channels, $\mathcal{E}_A(\cdot)$ and $\mathcal{E}_B(\cdot)$ denote learnable shallow convolutional networks designed to extract structural edge and texture features, $\{\mathbf{K}_i\}_{i=1}^{3}$ are fixed SRM high-pass filters, $\mathbf{W}_1$ and $\mathbf{W}_2$ are learnable convolutional kernels, BN_1 and BN_2 are batch-normalization layers, $\sigma(\cdot)$ denotes the ReLU activation, and ϵ is a small constant.

Feature Type	Feature Extractor $\phi_A^{(k)}$	Feature Extractor $\phi_B^{(k)}$
Spectral–Spatial Edge	$\phi_A^{(1)}(\mathbf{x}) \coloneqq \mathrm{Re}\left[\mathcal{F}^{-1}\left(\frac{\mathcal{F}(\mathcal{G}(\mathbf{x}))}{\lvert\mathcal{F}(\mathcal{G}(\mathbf{x}))\rvert+\epsilon}\right)\right]$	$\phi_B^{(1)}(\mathbf{x}) \coloneqq \sqrt{(\mathbf{S}_x * \mathcal{G}(\mathbf{x}))^2 + (\mathbf{S}_y * \mathcal{G}(\mathbf{x}))^2 + \epsilon}$
Luminance–Chrominance	$\phi_A^{(2)}(\mathbf{x}) \coloneqq \mathcal{E}_A(0.299\,\mathbf{x}_R + 0.587\,\mathbf{x}_G + 0.114\,\mathbf{x}_B)$	$\phi_B^{(2)}(\mathbf{x}) \coloneqq \mathcal{E}_B\left(\begin{bmatrix} -0.147\,\mathbf{x}_R - 0.289\,\mathbf{x}_G + 0.436\,\mathbf{x}_B \\ 0.615\,\mathbf{x}_R - 0.515\,\mathbf{x}_G - 0.100\,\mathbf{x}_B \end{bmatrix}\right)$
Noise–Texture	$\phi_A^{(3)}(\mathbf{x}) \coloneqq \begin{bmatrix} \sum_{c\in\{R,G,B\}} \mathbf{K}_1 * \mathbf{x} \\ \sum_{c\in\{R,G,B\}} \mathbf{K}_2 * \mathbf{x} \\ \sum_{c\in\{R,G,B\}} \mathbf{K}_3 * \mathbf{x} \end{bmatrix}$	$\phi_B^{(3)}(\mathbf{x}) \coloneqq \sigma(\mathrm{BN}_2(\mathbf{W}_2 * \sigma(\mathrm{BN}_1(\mathbf{W}_1 * \mathcal{G}(\mathbf{x})))))$

Building on the three physical cue dimensions introduced in Section 1—spectral-spatial edge structure, luminance-chrominance coupling, and noise-texture statistics—we construct three parallel feature extraction branches to extract related feature pairs for consistency evaluation, as shown in Fig. 2. Table 1 summarizes three paired feature extractors designed to capture the three aforementioned physical cues. For the k-th dimension, we denote the extracted feature pair as $\mathbf{z}_A^{(k)}$ and $\mathbf{z}_B^{(k)}$, obtained via $\phi_A^{(k)}$ and $\phi_B^{(k)}$, respectively. Then, we evaluate two types of violations of physics-aware feature consistency: (1) cross-channel inconsistencies in the global feature distribution, (2) element-wise absolute feature differences. To this end, we first project $\mathbf{z}_A^{(k)}$ and $\mathbf{z}_B^{(k)}$ into a shared latent space via two networks with the same architecture but separate parameters, ensuring comparability:

$$\hat{\mathbf{z}}_A^{(k)} \coloneqq \sigma_{\text{SiLU}}\left(\mathrm{IN}\left(\mathrm{Conv}_{3\times3}\left(\mathbf{z}_A^{(k)}\right)\right)\right), \quad \hat{\mathbf{z}}_B^{(k)} \coloneqq \sigma_{\text{SiLU}}\left(\mathrm{IN}\left(\mathrm{Conv}_{3\times3}\left(\mathbf{z}_B^{(k)}\right)\right)\right), \tag{3}$$

where $\mathrm{Conv}_{3\times3}(\cdot)$ denotes a 3×3 convolution, $\mathrm{IN}(\cdot)$ denotes instance normalization with learnable affine parameters, and $\sigma_{\mathrm{SiLU}}(\cdot)$ denotes the SiLU activation function. By designing an appropriate loss function (see Section 3.3), we make $\hat{\mathbf{z}}_A^{(k)}$ and $\hat{\mathbf{z}}_B^{(k)}$ comparable such that $\left|\hat{\mathbf{z}}_A^{(k)} - \hat{\mathbf{z}}_B^{(k)}\right|$ explicitly reflects the degree of physical-consistency violation. However, certain structural discrepancies in images are inherently implicit and require additional mechanisms to localize and emphasize the regions of violation. To this end, we define a *Physical Violation Map* (PVM) to capture spatially localized violations of physical consistency, given by

Table 2: Architecture of the fusion network $\mathcal{H}(\cdot)$, where all feature maps match the input resolution.

Layer	Operation	Mathematical Form
Input	Channel concatenation	$\hat{\mathbf{z}}_{\text{pair}} \coloneqq \hat{\mathbf{z}}_A^{(k)} \oplus \hat{\mathbf{z}}_B^{(k)}$
Conv-1	3×3 grouped convolution	$\mathbf{u}_1 = \mathbf{W}_1 * \hat{\mathbf{z}}_{\text{pair}}$
Conv-2	1×1 pointwise convolution	$\mathbf{u}_2 = \mathbf{W}_2 * \mathbf{u}_1$
Norm	Instance normalization	$\mathbf{u}_3 = \mathrm{IN}(\mathbf{u}_2)$
Act	SiLU activation	$\mathbf{u}_4 = \sigma_{\mathrm{SiLU}}(\mathbf{u}_3)$
Conv-3	1×1 projection	$\mathcal{H}(\cdot) = \mathbf{W}_3 * \mathbf{u}_4$

$$M_{\text{vio}}^{(k)} \coloneqq \mathrm{Sigmoid}\left(\mathrm{Conv}_{1\times1}\left(\mathcal{H}\left(\hat{\mathbf{z}}_A^{(k)} \oplus \hat{\mathbf{z}}_B^{(k)}\right)\right)\right), \tag{4}$$

where $\oplus$ denotes channel-wise concatenation, $\mathcal{H}$ denotes a multi-layer convolutional fusion network (detailed in Table 2) that integrates the channel-wise concatenation of $\hat{\mathbf{z}}_A^{(k)}$ and $\hat{\mathbf{z}}_B^{(k)}$ into a C-channel high-level structural inconsistency representation, $\mathrm{Conv}_{1\times1}(\cdot)$ denotes a 1×1 convolution that

reduces the C-channel feature map to a single channel, and $\mathrm{Sigmoid}(\cdot)$ denotes the sigmoid activation function, which constrains each element of $M_{\text{vio}}^{(k)}$ to the range $[0, 1]$ and represents the probability of physical-consistency violation. Since $M_{\text{vio}}^{(k)}$ shares the same spatial resolution as the input image, it can be viewed as a violation-aware weighting map that acts on feature maps through element-wise multiplication, emphasizing regions with strong physical inconsistencies. Accordingly, we construct the final discriminative representation $\mathbf{v}^{(k)}$ as

$$\mathbf{v}^{(k)} \coloneqq \underbrace{\left(\mathcal{H}\left(\hat{\mathbf{z}}_A^{(k)} \oplus \hat{\mathbf{z}}_B^{(k)}\right) \odot M_{\text{vio}}\right)}_{\text{cross-channel global inconsistency}} \oplus \underbrace{\left(\left|\hat{\mathbf{z}}_A^{(k)} - \hat{\mathbf{z}}_B^{(k)}\right| \odot M_{\text{vio}}\right)}_{\text{element-wise feature discrepancy}}, \tag{5}$$

where $\odot$ denotes the Hadamard (element-wise) product, $\oplus$ denotes channel-wise concatenation, and $\mathcal{H}(\cdot)$ denotes a learnable fusion network (detailed in Table 2) trained with a specialized loss to implicitly capture such violations (see Section 3.3). In contrast, the element-wise feature discrepancy term of Eq. (5) measures explicit differences between paired features. By jointly modeling implicit interactions and explicit discrepancies, Eq. (5) introduces complementary inductive biases that improve robustness against both coarse and subtle physical violations, as shown in Fig. 2.

3.3 Training and Prediction

Algorithm 1 Training Procedure of PhysCons

Input: training set $\mathcal{S}$, a learning rate η.
Output: learned parameters $\{\Theta_k\}_{k=1}^{3}$ of the detectors.
for k **in** $1, \ldots, 3$ **do**
  Initialize Θ_k for the k-th branch, including $\phi_A^{(k)}$, $\phi_B^{(k)}$, convolutional layers, normalization and activation layers, the fusion network $\mathcal{H}$, and the classification head.
  for iter **in** $1, \ldots, T$ **do**
    Sample a mini-batch $\mathbf{x}$ from $\mathcal{S}$;
    $\mathbf{z}_A^{(k)} \leftarrow \phi_A^{(k)}(\mathbf{x}), \quad \mathbf{z}_B^{(k)} \leftarrow \phi_B^{(k)}(\mathbf{x})$; ▷ See Table 1.
    $\hat{\mathbf{z}}_A^{(k)} \leftarrow \sigma_{\text{SiLU}}\left(\mathrm{IN}\left(\mathrm{Conv}_{3\times3}\left(\mathbf{z}_A^{(k)}\right)\right)\right)$;
    $\hat{\mathbf{z}}_B^{(k)} \leftarrow \sigma_{\text{SiLU}}\left(\mathrm{IN}\left(\mathrm{Conv}_{3\times3}\left(\mathbf{z}_B^{(k)}\right)\right)\right)$;
    $M_{\text{vio}}^{(k)} \leftarrow \mathrm{Sigmoid}\left(\mathrm{Conv}_{1\times1}\left(\mathcal{H}\left(\hat{\mathbf{z}}_A^{(k)} \oplus \hat{\mathbf{z}}_B^{(k)}\right)\right)\right)$;
    $\mathbf{f}^{(k)} \leftarrow \mathcal{H}\left(\hat{\mathbf{z}}_A^{(k)} \oplus \hat{\mathbf{z}}_B^{(k)}\right), \quad \mathbf{d}^{(k)} \leftarrow \left|\hat{\mathbf{z}}_A^{(k)} - \hat{\mathbf{z}}_B^{(k)}\right|$;
    $\mathbf{v}^{(k)} \leftarrow \left(\mathbf{f}^{(k)} \odot M_{\text{vio}}^{(k)}\right) \oplus \left(\mathbf{d}^{(k)} \odot M_{\text{vio}}^{(k)}\right)$;
    $\mathcal{L}_{\text{total}} \leftarrow \mathcal{L}_{\text{cls}} + \lambda\, \mathcal{L}_{\text{vio}}$; ▷ See Eqs. (6), (7).
    Compute gradients $\nabla \mathcal{L}_{\text{total}}$ via backpropagation;
    Update Θ_k using Adam with learning rate η;
  end for
end for

Since the proposed framework yields three physics-aware discriminative representations, $\mathbf{v}^{(1)}, \ldots, \mathbf{v}^{(3)}$, we train three independent branches for detection. If any branch predicts a fake image, we consider it fake. For the k-th feature type, the overall training loss is defined as a combination of multiple objective terms as $\mathcal{L}_{\text{total}} \coloneqq \mathcal{L}_{\text{cls}} + \lambda \cdot \mathcal{L}_{\text{vio}}$. The first term is a standard binary cross-entropy loss term, defined as

$$\mathcal{L}_{\text{cls}} \coloneqq -\frac{1}{N} \sum_{i=1}^{N} \Big[y_i \log \mathcal{P}(y = 1 \mid \mathbf{v}_i^{(k)}) + (1 - y_i) \log \big(1 - \mathcal{P}(y = 1 \mid \mathbf{v}_i^{(k)})\big) \Big], \tag{6}$$

where $\mathcal{P}(y = 1 \mid \mathbf{v}_i^{(k)}) \coloneqq \mathrm{Sigmoid}\left(\mathcal{C}^{(k)}(\mathbf{v}_i^{(k)})\right)$ denotes the predicted probability that the i-th sample is fake given its feature representation $\mathbf{v}_i^{(k)}$, $\mathcal{C}^{(k)}(\cdot)$ is a shallow classifier head composed of adaptive average pooling followed by two fully connected layers, N denotes the batch size, and y_i

denotes the label with 1 the fake sample and 0 the real sample. The violation loss $\mathcal{L}_{\text{vio}}$ is defined as

$$\mathcal{L}_{\text{vio}} \coloneqq \frac{1}{N}\sum_{i=1}^{N}\Big[(1-y_i)\cdot\|M_{\text{vio},i}^{(k)}\|_1 + y_i\cdot\max(0,\tau-\text{mean}(M_{\text{vio},i}^{(k)}))\Big], \tag{7}$$

where $M_{\text{vio},i}^{(k)}$ is the i-th sample's PVM. The objective of minimizing Eq. (7) is to induce $M_{\text{vio},i}^{(k)}$ with strong activations on fake images and near-zero responses on real images. This is implemented via a margin constraint for fake samples ($y_i = 1$) and an ℓ_1 penalty for real samples ($y_i = 0$).

The network is trained end-to-end, as summarized in Algorithm 1. For the k-th branch, the input image $\mathbf{x}$ is processed by two parallel sub-branches to extract a pair of features, which are subsequently fused to generate $M_{\text{vio}}^{(k)}$ and $\mathbf{v}^{(k)}$ for computing the loss. At test time, the processing pipeline remains identical to the training phase, i.e., the same feature extraction and fusion operations are applied to obtain $\mathbf{v}^{(k)}$. The only difference lies in the final decision stage, where we first aggregate the branch-wise predictions via the max-ensemble rule in Eq. (2), and then predict $\hat{y} = \mathbb{I}[\mathcal{P}_{\text{fake}}(\mathbf{x}) \geq 0.5]$.

4 Experiment

4.1 Experimental Settings

Datasets. To evaluate detector generalization, we conduct experiments on diverse datasets. First, we assess cross-generator generalization on GenImage (Zhu et al., 2023) (8 generators) and AIGCDetect-Benchmark (Zhong et al., 2023) (16 generators). All detectors are trained on a single generator—SD v1.4 (Rombach et al., 2022) for GenImage and ProGAN (Karras et al., 2018) for AIGCDetect-Benchmark—and evaluated on all remaining generators to assess generalization to unseen models. Second, we evaluate cross-dataset robustness to high-fidelity fake images collected online using the challenging Chameleon dataset (Yan et al., 2025a). Finally, to assess adaptability to emerging generators, we construct a new test set of 3,365 real and 3,365 fake images by extracting frames from collected real-world footage and Sora-generated videos (OpenAI, 2024).

Baselines. We benchmark our method against 12 representative open-source detectors, including CNNSpot, FreDect, Gram-Net, Fusing, LNP, LGrad, UnivFD, FatFormer, NPR, AF, AIDE, and Effort. These methods span a diverse range of technical paradigms, encompassing frequency-domain analysis, CLIP-based representations, image style statistics, and dataset alignment or reconstruction-based strategies, ensuring a thorough and fair comparative evaluation.

Metrics. In the main experiments, accuracy is reported as the primary metric, while precision, recall, and F1 score are additionally reported in the ablation studies.

Implementation Details. To ensure a fair comparison, we retrain baseline detectors following the method-specific settings provided in their official implementations. Across all methods, we use a batch size of 16 for both training and testing and train for 10 epochs. The checkpoint with the best validation performance is selected for final evaluation. Configurations are provided in Appendix B.

4.2 Comparison with State-of-the-Art Methods

Evaluation on standard benchmarks. We first evaluate cross-generator generalization on two widely used datasets, GenImage and AIGCDetectBenchmark, with the results summarized in Tables 3 and 4, respectively. To provide more granular insights into our framework, we report the performance of the three specialized physical modules—PhysCons-SS (spectral–spatial consistency), PhysCons-LC (luminance-chrominance coupling), and PhysCons-NT (noise–texture dependency)—as well as their maximum ensemble decision rule (Eq. (2)), denoted as PhysCons.

On the GenImage dataset (Table 3), where models are trained exclusively on SD v1.4, PhysCons achieves a strong average accuracy of 92.23%, outperforming the state-of-the-art baseline AIDE by 5.14 percentage points. Notably, PhysCons exhibits superior robustness across diverse generative models, particularly on challenging generators such as BigGAN (98.70%) and Midjourney (84.89%), where most baseline methods struggle to exceed 70% accuracy.

On the AIGCDetectBenchmark (Table 4), PhysCons-NT achieves the highest average accuracy of 92.84%, outperforming the strong baseline AIDE by 1.71 percentage points. Table 4 reveals a critical

Table 3: **Cross-generator evaluation on the GenImage dataset.** Accuracy (%) of different detectors trained on ImageNet (real) and SD v1.4 (fake), evaluated across eight generators. **Bold** and underline denote the best and second-best results, respectively.

Methods	Venue	SD v1.4	ADM	BigGAN	GLIDE	Midjourney	SD v1.5	VQDM	Wukong	Avg.
CNNSpot (Wang et al., 2020)	CVPR'20	99.62	50.96	50.35	53.23	62.08	99.64	50.22	98.77	70.61
FreDect (Frank et al., 2020)	ICML'20	98.99	50.17	49.59	53.76	62.28	98.89	49.55	96.53	69.97
Gram-Net (Liu et al., 2020)	CVPR'20	97.59	50.59	52.32	56.79	58.03	97.59	50.29	94.25	69.68
Fusing (Ju et al., 2022)	ICIP'22	99.86	54.77	51.14	61.29	65.87	99.80	69.74	99.81	75.28
LNP (Liu et al., 2022)	ECCV'22	99.87	49.61	49.64	49.71	50.58	99.77	49.65	99.76	68.57
LGrad (Tan et al., 2023)	CVPR'23	93.58	50.00	51.52	52.48	56.85	93.23	50.61	86.05	66.79
UnivFD (Ojha et al., 2023)	CVPR'23	96.80	54.98	66.83	82.35	82.05	96.91	56.10	91.63	78.46
FatFormer (Liu et al., 2024)	CVPR'24	99.93	52.12	56.45	61.79	56.18	99.83	54.91	98.59	72.48
NPR (Tan et al., 2024)	CVPR'24	**99.99**	58.10	50.50	77.60	75.00	99.60	53.50	97.40	76.46
AF (Sundara Rajan et al., 2025)	ICLR'25	99.89	50.56	52.13	51.67	52.13	99.87	57.46	**100.00**	70.46
AIDE (Yan et al., 2025a)	ICLR'25	99.97	**79.46**	67.17	93.07	79.49	99.85	78.23	99.46	87.09
Effort (Yan et al., 2025b)	ICML'25	**99.99**	71.20	76.28	72.13	79.50	**99.92**	**95.35**	99.63	86.75
PhysCons-SS (Ours)	-	97.83	49.95	96.57	85.50	58.69	97.62	51.16	94.60	78.99
PhysCons-LC (Ours)	-	97.83	68.47	97.92	96.12	77.22	99.85	72.99	98.39	88.60
PhysCons-NT (Ours)	-	98.23	66.59	84.57	79.47	70.33	97.94	72.50	94.58	83.03
PhysCons (Ours)	-	98.82	77.91	**98.70**	**97.06**	**84.89**	98.49	83.36	98.62	**92.23**

Table 4: **Cross-generator evaluation on the AIGCDetectBenchmark dataset.** Accuracy (%) of different detectors trained on LSUN (real) and ProGAN (fake), evaluated across sixteen generators. **Bold** and underline denote the best and second-best results, respectively.

Methods	*ProGAN*	*ADM*	*BigGAN*	*CycleGAN*	*DALLE2*	*GauGAN*	*GLIDE*	*Midjourney*	*SD v1.4*	*SD v1.5*	*StarGAN*	*StyleGAN*	*StyleGAN2*	*VQDM*	*WFIR*	*Wukong*	*Avg.*
CNNSpot (Wang et al., 2020)	99.86	56.73	63.50	80.89	50.60	71.60	54.13	51.90	51.70	52.01	84.39	82.78	79.79	51.88	62.95	50.62	65.33
FreDect (Frank et al., 2020)	99.75	60.67	67.83	53.29	30.55	62.61	47.59	33.27	27.93	27.97	92.87	75.85	68.44	68.34	43.80	37.24	56.12
Gram-Net (Liu et al., 2020)	99.64	53.75	69.43	77.06	52.65	69.78	49.91	53.98	53.02	53.32	75.51	91.12	95.67	58.42	63.15	48.99	66.59
Fusing (Ju et al., 2022)	99.96	83.98	66.15	81.00	**95.35**	52.78	79.87	60.96	91.22	91.03	95.50	83.48	83.30	79.68	55.10	85.03	80.27
LNP (Liu et al., 2022)	99.78	81.55	78.30	62.53	93.65	71.38	81.65	56.37	50.76	50.57	99.85	78.40	83.26	65.72	41.75	50.04	71.60
LGrad (Tan et al., 2023)	98.81	61.17	74.03	74.15	57.45	65.57	60.61	61.53	61.42	61.64	93.75	94.09	92.63	62.76	57.25	58.71	70.97
UnivFD (Ojha et al., 2023)	99.94	72.18	96.23	**98.18**	51.25	**99.17**	66.65	50.24	52.78	52.74	95.02	84.50	77.05	88.92	81.20	57.89	76.50
FatFormer (Liu et al., 2024)	**99.99**	60.47	77.35	84.52	51.20	66.95	55.58	56.14	64.69	65.32	**100.00**	88.14	95.01	59.33	52.55	60.61	71.12
NPR (Tan et al., 2024)	**99.99**	66.67	79.20	82.78	64.80	72.76	77.74	79.89	79.99	80.89	95.47	92.28	96.03	62.98	63.10	73.72	79.27
AF (Sundara Rajan et al., 2025)	**99.99**	68.44	96.98	94.10	50.63	96.04	59.18	51.07	52.19	52.01	99.22	99.10	90.72	90.72	**93.35**	55.23	78.06
AIDE (Yan et al., 2025a)	99.78	90.39	81.36	92.26	93.11	71.15	92.37	76.63	91.92	92.13	99.94	**99.31**	**99.02**	93.85	92.63	92.23	91.13
Effort (Yan et al., 2025b)	**99.99**	75.27	**99.39**	97.72	81.55	96.37	86.94	56.13	90.77	89.92	**100.00**	96.63	96.91	86.08	86.73	84.69	89.07
PhysCons-SS (Ours)	96.10	66.78	86.50	86.56	72.00	89.83	86.11	63.93	80.67	80.33	98.27	83.43	82.79	82.21	54.35	77.82	80.48
PhysCons-LC (Ours)	97.55	82.04	92.00	87.09	87.15	97.68	90.04	78.81	92.24	92.00	99.27	95.57	89.10	88.30	63.40	86.42	88.67
PhysCons-NT (Ours)	99.84	**90.67**	94.70	94.17	90.40	96.33	**96.35**	**91.07**	**94.15**	**94.39**	**100.00**	95.75	97.30	**95.45**	62.10	**92.76**	**92.84**
PhysCons (Ours)	96.64	83.64	85.95	91.26	83.10	89.50	84.62	81.70	84.44	84.40	98.92	88.21	89.87	84.27	56.15	84.41	85.44

phenomenon: while most existing methods perform well on GAN-based generators, they suffer substantial performance degradation on diffusion-based models (e.g., UnivFD drops from 98.18% on CycleGAN (Zhu et al., 2017) to 52.78% on SD v1.4). In contrast, our PhysCons-NT maintains robust performance across both paradigms (e.g., 100% on StarGAN (Choi et al., 2018) and 94.15% on SD v1.4). These results indicate that our physical modules capture generator-agnostic cues that generalize across both GAN-based and diffusion-based models, rather than relying on model- or modality-specific statistical artifacts. For the full ensemble, max aggregation preserves all branch detections, ensuring recall no lower than that of any individual branch to prioritize the detection of forged images while trading off accuracy.

Evaluation on challenging and realistic scenarios. To assess real-world robustness, we evaluate on Chameleon and a newly constructed Sora test set targeting state-of-the-art video generation. Table 5 shows that most existing detectors achieve average accuracies of only 50%–55%, marginally above random guessing, indicating limited generalization beyond the training distribution. In contrast, PhysCons generalizes strongly across these challenging domains. PhysCons-NT performs best on Chameleon (69.92%), while PhysCons-SS achieves the highest accuracy on Sora (75.54%), demonstrating effectiveness in generated video detection. Overall, our method achieves 71.43% average accuracy, exceeding the strongest baseline, LNP, by 15.89 percentage points and demonstrating the universality of physical inconsistencies across diverse generative paradigms, including video models such as Sora. Further results on emerging generators are reported in Appendix C.3.

Computational efficiency analysis. We evaluate the computational efficiency of all methods in terms of inference speed (frames per second, FPS) and model parameters (Params). As shown in Table 6,

Table 5: The evaluation on challenging and realistic scenarios. Accuracy (%) of different detectors evaluated on Chameleon and Sora. All models are trained exclusively on ProGAN.

Source	Methods															
	CNNSpot	FreDect	Fusing	LNP	LGrad	Gram-Net	UnivFD	FatFormer	AF	AIDE	NPR	Effort	PhysCons -SS (Ours)	PhysCons -LC (Ours)	PhysCons -NT (Ours)	PhysCons (Ours)
Chameleon	56.84	55.24	58.48	58.97	57.63	57.00	56.25	60.22	56.72	57.39	59.89	58.94	64.06	57.93	**69.92**	<u>68.64</u>
Sora	51.66	53.31	52.11	52.11	45.27	43.28	44.34	41.17	52.11	51.08	42.21	51.44	**75.54**	<u>74.78</u>	72.93	73.89
Average	54.25	54.28	55.30	55.54	51.45	50.14	50.30	50.70	54.42	54.24	51.05	55.19	69.80	66.36	**71.43**	<u>71.27</u>

Table 6: Comparison of computational efficiency of various detectors, measured by inference speed (frames per second, FPS) and model parameters (in millions). All methods are evaluated on the Chameleon dataset using an NVIDIA RTX 4090 GPU.

Metrics	Methods															
	CNNSpot	FreDect	Fusing	LNP	LGrad	Gram-Net	UnivFD	FatFormer	AF	AIDE	NPR	Effort	PhysCons -SS (Ours)	PhysCons -LC (Ours)	PhysCons -NT (Ours)	PhysCons (Ours)
FPS ↑	294	341	35	289	222	369	114	39	102	52	562	142	**1378**	<u>1301</u>	1297	722
Params (M) ↓	23.51	23.51	26.12	23.51	23.51	11.73	427.60	492.59	23.51	53.43	1.44	124.63	**0.01**	**0.01**	<u>0.03</u>	0.05

our framework exhibits a clear efficiency advantage. Thanks to its lightweight design, PhysCons achieves an inference speed of 722 FPS, which is 1.28× faster than the efficient NPR method (562 FPS) and 13.88× faster than the heavy-weight AIDE model (52 FPS). Moreover, PhysCons requires only 0.05M parameters in total, which is orders of magnitude fewer than strong baselines such as UnivFD (427.60M) and FatFormer (492.59M). This favorable trade-off between detection accuracy and computational efficiency makes our method well suited for large-scale and real-time deployment.

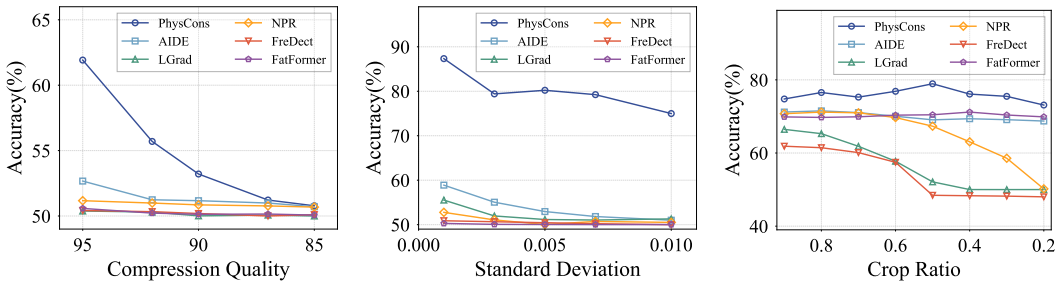
4.3 Evaluation on Robustness to Image Post-Processing

To evaluate the robustness of the PhysCons framework against real-world image post-processing operations, we subject the test images—comprising both real and generated samples—to three common perturbations: (1) JPEG compression with multiple quality factors; (2) Gaussian noise perturbations with varying noise levels; (3) random cropping with crop ratios ranging from 0.9 to 0.2. To ensure a fair comparison, all compared methods are trained exclusively on the SD v1.4 subset of the GenImage dataset and evaluated across all eight generators to report the performance. Furthermore, the same data augmentation strategy is used for all models during training to ensure a fair comparison. The experimental results are illustrated in Fig. 3. Under JPEG compression with varying quality factors, PhysCons consistently achieves higher detection accuracy than baseline methods, demonstrating strong robustness to the loss of high-frequency information induced by compression. Similarly, under Gaussian noise perturbations, our method exhibits notable stability, with performance degrading much more gracefully than that of competing approaches. Furthermore, PhysCons maintains stable detection performance under random cropping, where its accuracy degradation is substantially smaller than that of baseline methods as the crop ratio decreases from 0.9 to 0.2, showing its robustness.

4.4 Comprehensive Understanding of PhysCons

In the ablation studies, we first investigate the effectiveness of physical feature combinations described in Table 1, with results in Table 7. We evaluate the three physical consistency modules individually, in pairs, and as a full ensemble. Each pairwise combination improves accuracy, recall, and F1 score over its constituent branches, highlighting the complementary value of different physical consistency cues. The full PhysCons ensemble also outperforms every individual branch on these three metrics and achieves the highest recall (88.42%) among all evaluated configurations, aligning with our goal of minimizing missed detections.

Second, we evaluate the necessity of the loss function $\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{cls}} + \lambda \cdot \mathcal{L}_{\text{vio}}$ by removing the violation loss $\mathcal{L}_{\text{vio}}$ (Eq. (7)) and retraining the models. Table 8 summarizes the accuracy performance under different loss settings. Removing $\mathcal{L}_{\text{vio}}$ consistently degrades the performance of all PhysCons branches, with accuracy drops ranging from 0.36% to 5.30%, highlighting the importance of $\mathcal{L}_{\text{vio}}$.



(a) JPEG compression (b) Gaussian noise perturbations (c) Random cropping

Figure 3: Robustness to image post-processing: (a) **JPEG compression**, quality factors $95 \rightarrow 85$; (b) **Gaussian noise**, standard deviations $0.001 \rightarrow 0.01$; (c) **random cropping**, crop ratios $0.9 \rightarrow 0.2$.

Table 7: Results of ablation studies on different physical feature combinations. We compare individual modules (SS: Spectral-Spatial, LC: Luminance-Chrominance, NT: Noise-Texture), pairwise combinations, and the full PhysCons framework using Accuracy (Acc), Precision (Pre), Recall (Rec), and F1 score (F1). All metrics are computed independently on each GenImage subset and then averaged; thus, F1 is not derived from the averaged precision and recall.

Combination of branches				Metrics (%)			
Model	SS	LC	NT	Acc	Pre	Rec	F1
Individual	✓	×	×	78.99	84.57	61.55	66.18
	×	✓	×	88.60	96.71	79.01	85.51
	×	×	✓	83.03	96.15	68.00	78.06
Pairwise	✓	✓	×	89.81	94.94	83.34	87.53
	✓	×	✓	88.27	94.48	80.35	85.33
	×	✓	✓	**92.82**	**97.21**	87.73	**91.70**
PhysCons	✓	✓	✓	92.23	95.36	**88.42**	91.27

Table 8: Loss ablation. Accuracy (%) with classification loss alone ($\mathcal{L}_{\text{cls}}$) and the full objective ($\mathcal{L}_{\text{cls}} + \lambda\mathcal{L}_{\text{vio}}$).

Method	$\mathcal{L}_{\text{cls}}$	$\mathcal{L}_{\text{cls}} + \lambda\mathcal{L}_{\text{vio}}$	Gain
PhysCons-SS	78.63	78.99	0.36
PhysCons-LC	85.32	88.60	3.28
PhysCons-NT	77.73	83.03	5.30
PhysCons	90.01	92.23	2.22

Table 9: Ablation of the discriminative representation. Accuracy (%) with and without the global inconsistency term in $\mathbf{v}^{(k)}$ defined in Eq. (5).

Method	w/o $\mathcal{H}(\cdot)$	Full (Eq. (5))	Gain
PhysCons-SS	76.99	78.99	2.00
PhysCons-LC	83.11	88.60	5.49
PhysCons-NT	82.60	83.03	0.43
PhysCons	84.98	92.23	7.25

Finally, we investigate the contribution of each component in the discriminative representation $\mathbf{v}^{(k)}$ defined in Eq. (5). As reported in Table 9, removing the cross-channel global inconsistency term, i.e., $\left(\mathcal{H}\left(\hat{\mathbf{z}}_A^{(k)} \oplus \hat{\mathbf{z}}_B^{(k)}\right) \odot M_{\text{vio}}\right)$, consistently degrades detection accuracy across all PhysCons variants. The performance drop is particularly pronounced under the ensemble decision rule (PhysCons), where accuracy decreases by 7.25%. This observation confirms that modeling global inconsistency plays a critical role in enhancing the discriminative power of the final representation.

5 Conclusion

We propose PhysCons, a physics-aware framework that detects AI-generated images by modeling intrinsic physical inconsistencies. By modeling cross-signal physical consistency, PhysCons generalizes well to unseen generators, including Stable Diffusion and video generation models. Despite using only 0.05M parameters, PhysCons achieves state-of-the-art performance at 722 FPS, highlighting the effectiveness of physics-grounded detection.

AI USE STATEMENT

In this work, we used generative AI tools to generate synthetic data for evaluating AI-generated image detection. Additionally, we used generative AI tools to polish the manuscript, identify grammatical errors and assist with LaTeX formatting. Their use was limited to these purposes; we did not use them to develop the proposed method, implement detection algorithms, or interpret experimental results. We reviewed all AI-assisted text for technical accuracy and checked the generated evaluation data for consistency with the intended generation settings and dataset labels. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

PhysCons is intended to support visual authenticity assessment and mitigate the misuse of AI-generated imagery. Nevertheless, false positives may undermine trust in authentic content, while false negatives may allow synthetic content to evade detection. Its predictions and physical violation maps should therefore serve as supporting evidence rather than conclusive proof of an image's origin or an individual's intent. Deployment should involve human review and validation across the intended image domains and populations. Our experiments use existing benchmarks, collected image and video data, and generated evaluation samples. The use and any redistribution of these materials should respect applicable licenses, platform terms, copyright, and privacy, particularly for content depicting identifiable individuals. We also acknowledge the potential misuse of the proposed consistency cues to develop detection-evasion strategies. Appendix A discusses limitations relevant to responsible deployment, including false positives on processed photographs and difficulties in detecting localized edits.

REPRODUCIBILITY STATEMENT

Section 3 describes the PhysCons framework, including the paired feature extractors in Table 1, the fusion architecture in Table 2, and the training procedure in Algorithm 1. Section 3.3 specifies the training objectives and prediction rule. Section 4.1 and Appendix B describe the datasets, evaluation protocols, preprocessing, hardware, and implementation settings. Method-specific configurations are summarized in Table 12, and Appendix B details the collection, frame sampling, and deduplication procedure for the Sora-based evaluation set. Section 4.4 and Appendix C provide ablations and additional evaluations, while Appendix D presents complementary visual analyses. The implementation code will be submitted as supplementary material to facilitate independent verification and reproduction.

REFERENCES

Black Forest Labs. FLUX. https://github.com/black-forest-labs/flux, 2024. Accessed: 2025-08-01.

Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale GAN training for high fidelity natural image synthesis. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=B1xsqj09Fm.

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 *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10759–10769, 2024. doi: 10.1109/CVPR52733.2024.01023.

Baoying Chen, Jishen Zeng, Jianquan Yang, and Rui Yang. DRCT: Diffusion reconstruction contrastive training towards universal detection of diffusion generated images. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 7621–7639. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/chen24ay.html.

Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. StarGAN: Unified generative adversarial networks for multi-domain image-to-image translation. In *2018*

IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 8789–8797, 2018. doi: 10.1109/CVPR.2018.00916.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, 2009. doi: 10.1109/CVPR.2009.5206848.

Prafulla Dhariwal and Alexander Nichol. Diffusion models beat GANs on image synthesis. In *Advances in Neural Information Processing Systems*, volume 34, pp. 8780–8794. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/49ad23d1ec9fa4bd8d77d02681df5cfa-Paper.pdf.

Ione Fine, Donald IA MacLeod, and Geoffrey M Boynton. Surface segmentation based on the luminance and color statistics of natural scenes. *Journal of the Optical Society of America A*, 20 (7):1283–1291, 2003.

Joel Frank, Thorsten Eisenhofer, Lea Schönherr, Asja Fischer, Dorothea Kolossa, and Thorsten Holz. Leveraging frequency analysis for deep fake image recognition. In *Proceedings of the International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 3247–3258. PMLR, 2020. URL https://proceedings.mlr.press/v119/frank20a.html.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Commun. ACM*, 63(11): 139–144, October 2020. ISSN 0001-0782. doi: 10.1145/3422622. URL https://doi.org/10.1145/3422622.

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 *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10686–10696, 2022. doi: 10.1109/CVPR52688.2022.01043.

Thorsten Hansen and Karl R Gegenfurtner. Independence of color and luminance edges in natural scenes. *Visual neuroscience*, 26(1):35–49, 2009.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, pp. 6840–6851. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/4c5bcfec8584af0d967f1ab10179ca4b-Paper.pdf.

Huawei. Huawei honors Wukong-Huahua sci-fi AI fine-tuning and application competition winners, 2023. URL https://www.huawei.com/en/news/2023/10/chengdumindspore-aiworldcon.

Yan Ju, Shan Jia, Lipeng Ke, Hongfei Xue, Koki Nagano, and Siwei Lyu. Fusing global and local features for generalized AI-synthesized image detection. In *2022 IEEE International Conference on Image Processing (ICIP)*, pp. 3465–3469, 2022. doi: 10.1109/ICIP46576.2022.9897820.

Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of GANs for improved quality, stability, and variation. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=Hk99zCeAb.

Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of StyleGAN. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8107–8116, 2020. doi: 10.1109/CVPR42600.2020.00813.

Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 43(12): 4217–4228, 2021. ISSN 1939-3539. doi: 10.1109/TPAMI.2020.2970919. URL https://doi.ieeecomputersociety.org/10.1109/TPAMI.2020.2970919.

Peter Kovesi. Image features from phase congruency. *Videre: Journal of computer vision research*, 1 (3):1–26, 1999.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common objects in context. In *European Conference on Computer Vision (ECCV)*, pp. 740–755, 2014.

Bo Liu, Fan Yang, Xiuli Bi, Bin Xiao, Weisheng Li, and Xinbo Gao. Detecting generated images by real images. In *Proceedings of the European Conference on Computer Vision*, pp. 95–110, Berlin, Heidelberg, 2022. Springer-Verlag. doi: 10.1007/978-3-031-19781-9_6.

Huan Liu, Zichang Tan, Chuangchuang Tan, Yunchao Wei, Jingdong Wang, and Yao Zhao. Forgery-aware adaptive transformer for generalizable synthetic image detection. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10770–10780, 2024. doi: 10.1109/CVPR52733.2024.01024.

Xinhao Liu, Masayuki Tanaka, and Masatoshi Okutomi. Practical signal-dependent noise parameter estimation from a single noisy image. *IEEE Transactions on Image Processing*, 23(10):4361–4371, 2014. doi: 10.1109/TIP.2014.2347204.

Zhengzhe Liu, Xiaojuan Qi, and Philip H.S. Torr. Global texture enhancement for fake face detection in the wild. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8057–8066, 2020. doi: 10.1109/CVPR42600.2020.00808.

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 3730–3738, 2015. doi: 10.1109/ICCV.2015.425.

Yunpeng Luo, Junlong Du, Ke Yan, and Shouhong Ding. LaRE^2: Latent reconstruction error based method for diffusion-generated image detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17006–17015, 2024. doi: 10.1109/CVPR 52733.2024.01609.

Midjourney. Midjourney, 2023. URL `https://www.midjourney.com/home`. Accessed: 2023-04-11.

M Concetta Morrone and Robyn A Owens. Feature detection from local energy. *Pattern recognition letters*, 6(5):303–313, 1987.

Alexander Quinn Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob Mcgrew, Ilya Sutskever, and Mark Chen. GLIDE: Towards photorealistic image generation and editing with text-guided diffusion models. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 16784–16804. PMLR, 17–23 Jul 2022. URL `https://proceedings.mlr.press/v162/nichol22a.html`.

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. doi: 10.1109/CVPR52729.2023.02345.

OpenAI. Video generation models as world simulators. `https://openai.com/research/video-generation-models-as-world-simulators`, 2024.

Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2332–2341, 2019. doi: 10.1109/CVPR.2019.00244.

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 Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763. PMLR, 18–24 Jul 2021. URL `https://proceedings.mlr.press/v139/radford21a.html`.

Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text conditional image generation with CLIP latents, 2022. URL `https://arxiv.org/abs/2204.06125`.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10674–10685, 2022. doi: 10.1109/CVPR52688.2022.01042.

Bernhard Schölkopf, Francesco Locatello, Stefan Bauer, Nan Rosemary Ke, Nal Kalchbrenner, Anirudh Goyal, and Yoshua Bengio. Toward causal representation learning. *Proceedings of the IEEE*, 109(5):612–634, 2021. doi: 10.1109/JPROC.2021.3058954.

Anirudh Sundara Rajan, Utkarsh Ojha, Jedidiah Schloesser, and Yong Jae Lee. Aligned datasets improve detection of latent diffusion-generated images. In *International Conference on Learning Representations*, volume 2025, pp. 63304–63322, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/9ead108421b202494d01b5060d12aa34-Paper-Conference.pdf.

Chuangchuang Tan, Yao Zhao, Shikui Wei, Guanghua Gu, and Yunchao Wei. Learning on gradients: Generalized artifacts representation for GAN-generated images detection. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12105–12114, 2023. doi: 10.1109/CVPR52729.2023.01165.

Chuangchuang Tan, Huan Liu, 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 *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 28130–28139, 2024. doi: 10.1109/CVPR52733.2024.02657.

Sheng-Yu Wang, Oliver Wang, Richard Zhang, Andrew Owens, and Alexei A. Efros. CNN-generated images are surprisingly easy to spot... for now. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8692–8701, 2020. doi: 10.1109/CVPR42600.2020.00872.

Zhendong Wang, Jianmin Bao, Wengang Zhou, Weilun Wang, Hezhen Hu, Hong Chen, and Houqiang Li. DIRE for diffusion-generated image detection. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 22388–22398, 2023. doi: 10.1109/ICCV51070.2023.02051.

Jevin West and Carl Bergstrom. Which Face Is Real? https://www.whichfaceisreal.com/, 2019.

Shilin Yan, Ouxiang Li, Jiayin Cai, Yanbin Hao, Xiaolong Jiang, Yao Hu, and Weidi Xie. A sanity check for AI-generated image detection. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 70702–70720, 2025a. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/b0303773962ea1b5394c3a83cc7dd066-Paper-Conference.pdf.

Zhiyuan Yan, Jiangming Wang, Peng Jin, Ke-Yue Zhang, Chengchun Liu, Shen Chen, Taiping Yao, Shouhong Ding, Baoyuan Wu, and Li Yuan. Orthogonal subspace decomposition for generalizable AI-generated image detection. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 70268–70288. PMLR, 13–19 Jul 2025b. URL https://proceedings.mlr.press/v267/yan25b.html.

Zhiyuan Yan, Junyan Ye, Weijia Li, Zilong Huang, Shenghai Yuan, Xiangyang He, Kaiqing Lin, Jun He, Conghui He, and Li Yuan. GPT-ImgEval: A comprehensive benchmark for diagnosing GPT4o in image generation, 2025c. URL https://arxiv.org/abs/2504.02782.

Fisher Yu, Ari Seff, Yinda Zhang, Shuran Song, Thomas Funkhouser, and Jianxiong Xiao. LSUN: Construction of a large-scale image dataset using deep learning with humans in the loop, 2015. URL https://arxiv.org/abs/1506.03365.

Nan Zhong, Yiran Xu, Sheng Li, Zhenxing Qian, and Xinpeng Zhang. PatchCraft: Exploring texture patch for efficient AI-generated image detection, 2023. URL https://arxiv.org/abs/2311.12397.

Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 2242–2251, 2017. doi: 10.1109/ICCV.2017.244.

Mingjian Zhu, Hanting Chen, Qiangyu YAN, Xudong Huang, Guanyu Lin, Wei Li, Zhijun Tu, Hailin Hu, Jie Hu, and Yunhe Wang. GenImage: A million-scale benchmark for detecting AI-generated image. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 77771–77782. Curran Associates, Inc., 2023. doi: 10.52202/075280-3398. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/f4d4a021f9051a6c18183b059117e8b5-Paper-Datasets_and_Benchmarks.pdf.

APPENDIX

A LIMITATIONS

Although PhysCons shows strong benchmark performance, several challenges remain in real-world deployment. Modern computational photography operations, such as facial beautification, may alter real images and introduce generation-like traces, increasing false positives. Moreover, small localized AI edits can produce sparse artifacts that are difficult to capture with image-level detectors. Finally, the proposed physical-consistency cues may be exploited by adversarial training to evade detection, calling for more fine-grained and adversarially robust detection mechanisms in future work.

B EXPERIMENTAL DETAILS

In this section, we provide the hyperparameter settings for our approach and the compared methods, which include CNNSpot (Wang et al., 2020), FreDect (Frank et al., 2020), Gram-Net (Liu et al., 2020), Fusing (Ju et al., 2022), LNP (Liu et al., 2022) ,LGrad (Tan et al., 2023), UnivFD (Ojha et al., 2023), FatFormer (Liu et al., 2024), NPR (Tan et al., 2024), AF (Sundara Rajan et al., 2025), AIDE (Yan et al., 2025a), and Effort (Yan et al., 2025b).

Experimental Equipment. The experiments of all methods are conducted on NVIDIA GeForce RTX 4090 and NVIDIA GeForce RTX 2080 Ti GPUs. The NVIDIA GeForce RTX 4090 GPU features TensorFloat-32 (TF32) tensor cores to improve computation speed; however, enabling TF32 tensor cores may cause a large relative error compared to double precision. Therefore, in all experiments, we set `torch.backends.cuda.matmul.allowtf32 = False` and `torch.backends.cudnn.allowtf32 = False` to ensure higher numerical precision.

Table 10: **Detailed statistics of the GenImage dataset.** The dataset utilizes images generated from ImageNet classes. We employ Stable Diffusion v1.4 (Train) for training and evaluate generalization across eight diverse testing subsets, categorized by their underlying generative architectures (GANs vs. Diffusion Models).

Family	Generator	Resolution	Source Domain
GAN	BigGAN (Brock et al., 2019)	256×256	ImageNet (Deng et al., 2009)
Diffusion	Stable Diffusion v1.4 (Train) (Rombach et al., 2022)	512×512	ImageNet (Deng et al., 2009)
	ADM (Dhariwal & Nichol, 2021)	256×256	ImageNet (Deng et al., 2009)
	GLIDE (Nichol et al., 2022)	256×256	ImageNet (Deng et al., 2009)
	Midjourney (Midjourney, 2023)	1024×1024	ImageNet (Deng et al., 2009)
	Stable Diffusion v1.4 (Test) (Rombach et al., 2022)	512×512	ImageNet (Deng et al., 2009)
	Stable Diffusion v1.5 (Rombach et al., 2022)	512×512	ImageNet (Deng et al., 2009)
	VQDM (Gu et al., 2022)	256×256	ImageNet (Deng et al., 2009)
	Wukong (Huawei, 2023)	512×512	ImageNet (Deng et al., 2009)

Dataset Details. To evaluate the robustness of our method, we conduct experiments on two recognized benchmarks: **GenImage** (Zhu et al., 2023) and **AIGCDetectBenchmark** (Zhong et al., 2023). These datasets cover diverse architectures, including GANs and DDPMs. GenImage focuses on diffusion models, utilizing images primarily from ImageNet (Deng et al., 2009) (Table 10). Following the official protocol, we use **Stable Diffusion v1.4** for training (324k real/fake pairs). The test set involves eight generators, including Midjourney (Midjourney, 2023), GLIDE (Nichol et al., 2022), and Wukong (Huawei, 2023), challenging the detector to generalize to unseen architectures. We also employ AIGCDetectBenchmark for cross-family evaluation (Table 11). Its training set uses **ProGAN** (Karras et al., 2018) on LSUN. The extensive test set comprises 16 models across various resolutions (from 256×256 to 1024×1024) and domains (e.g., FFHQ, LSUN, COCO). This verifies if our method, trained on GAN data, generalizes to diffusion models like DALLE 2 (Ramesh et al., 2022) and Stable Diffusion (Rombach et al., 2022).

To further simulate real-world scenarios and evaluate the robustness of detectors against high-fidelity samples, we also employ the Chameleon dataset (Yan et al., 2025a). The AI-generated images in this dataset are sourced from professional AI art communities and encompass a variety of advanced generators, including Midjourney, DALL·E 3, and LoRA-finetuned variants of Stable Diffusion.

Table 11: **Overview of the AIGCDetectBenchmark dataset.** This benchmark serves as a comprehensive evaluation platform containing 16 distinct generative models. It includes a training set generated by ProGAN and a wide array of testing sets covering various architectures (GANs and Diffusion Models) and source domains (e.g., LSUN, CelebA, FFHQ).

Family	Generator	Resolution	Source Domain
GAN	ProGAN (Train) (Karras et al., 2018)	256×256	LSUN (Yu et al., 2015)
	ProGAN (Test) (Karras et al., 2018)	256×256	LSUN (Yu et al., 2015)
	StyleGAN (Karras et al., 2021)	256×256	LSUN (Yu et al., 2015)
	BigGAN (Brock et al., 2019)	256×256	ImageNet (Deng et al., 2009)
	CycleGAN (Zhu et al., 2017)	256×256	ImageNet (Deng et al., 2009)
	StarGAN (Choi et al., 2018)	256×256	CelebA (Liu et al., 2015)
	GauGAN (Park et al., 2019)	256×256	COCO (Lin et al., 2014)
	StyleGAN2 (Karras et al., 2020)	256×256	LSUN (Yu et al., 2015)
	WFIR (West & Bergstrom, 2019)	1024×1024	FFHQ (Karras et al., 2021)
Diffusion	ADM (Dhariwal & Nichol, 2021)	256×256	ImageNet (Deng et al., 2009)
	GLIDE (Nichol et al., 2022)	256×256	ImageNet (Deng et al., 2009)
	Midjourney (Midjourney, 2023)	1024×1024	ImageNet (Deng et al., 2009)
	Stable Diffusion v1.4 (Rombach et al., 2022)	512×512	ImageNet (Deng et al., 2009)
	Stable Diffusion v1.5 (Rombach et al., 2022)	512×512	ImageNet (Deng et al., 2009)
	VQDM (Gu et al., 2022)	256×256	ImageNet (Deng et al., 2009)
	Wukong (Huawei, 2023)	512×512	ImageNet (Deng et al., 2009)
	DALLE 2 (Ramesh et al., 2022)	256×256	ImageNet (Deng et al., 2009)

Crucially, the dataset adheres to a strict "Turing Test" protocol: only images that successfully deceived human annotators (i.e., were misclassified as real) or were rated as highly realistic are included. Chameleon comprises approximately $26{,}000$ images, spanning four diverse categories—Scene, Object, Animal, and Human—with resolutions ranging from 720P to 4K, making it significantly more challenging than existing datasets.

To evaluate the generalization capability of detectors against state-of-the-art text-to-video models, we constructed a supplementary benchmark focused on Sora (OpenAI, 2024), as shown in Fig. 4. We collected 168 publicly available Sora-generated videos and 150 high-quality real-world videos from online platforms, covering a wide range of scenarios. To adapt these video data for image detection, we performed temporal downsampling at a rate of 1 frame per second. We then applied perceptual hashing to eliminate duplicate or highly similar frames. From the resulting pool, we curated a balanced dataset consisting of $3{,}365$ AI-generated frames and $3{,}365$ real-world frames, ensuring rigorous evaluation.

Baseline Settings. Table 12 summarizes the implementation details for all baseline methods and our approach. To ensure a fair comparison under uniform hardware constraints, we standardize the batch size to 16 across all experiments. However, for other hyperparameters—including learning rates, specific data augmentation pipelines, and normalization statistics—we strictly adhere to the official open-source implementations to respect the original design of each method.

C Additional Results and Analysis

In the main manuscript, we report Accuracy (Acc) to demonstrate the practical detection performance under standard decision thresholds, which directly reflects real-world deployment scenarios. To provide a more comprehensive evaluation, we further report ROC-AUC scores in this supplementary section. Since ROC-AUC is independent of a particular threshold, these results verify whether the performance gain of PhysCons comes from intrinsically separable representations rather than threshold tuning. Table 13 and Table 14 detail these comparisons on the GenImage and AIGCDetectBenchmark datasets, respectively.

C.1 ROC-AUC Evaluation on GenImage

Table 13 reports the ROC-AUC scores on the GenImage dataset, where all detectors are trained on Stable Diffusion v1.4 and evaluated across eight generators. PhysCons consistently achieves strong

Table 12: **Implementation details and pre-processing settings for baseline methods and our approach.** We summarize the batch size (BS), learning rate (LR), and specific data augmentation/normalization pipelines used in our experiments.

Method	Description	Hyperparameters & Pre-processing
CNNSpot	Uses simple augmentations (JPEG/Blur) to generalize to unknown GAN architectures.	**BS**: 16 **LR**: 10^{-4} **Input**: Resize 256 → Random Crop 224 **Aug**: Gaussian Blur (10%), JPEG Compression (10%), and Random Horizontal Flip (50%) **Norm**: ImageNet Stats; Mean: [0.485, 0.456, 0.406], Std: [0.229, 0.224, 0.225]
FreDect	Exploits structural artifacts in the DCT domain caused by GAN upsampling.	**BS**: 16 **LR**: 10^{-4} **Input**: Resize 256 → Random Crop 224 → 2D-DCT (Log scale) **Norm**: Custom DCT Mean/Var
Fusing	Fuses global context with local patches selected via a self-supervised module.	**BS**: 16 **LR**: 10^{-4} **Input**: Global branch uses original image; local branch uses Resize 256 → Crop 224 **Norm**: ImageNet Stats; Mean: [0.485, 0.456, 0.406], Std: [0.229, 0.224, 0.225]
Gram-Net	Leverages global texture analysis via Gram matrices.	**BS**: 16 **LR**: 10^{-4} **Input**: Resize 256 → Random Crop 224 **Aug**: Gaussian Blur (10%), JPEG Compression (10%), and Random Horizontal Flip (50%) **Norm**: ImageNet Stats; Mean: [0.485, 0.456, 0.406], Std: [0.229, 0.224, 0.225]
LGrad	Uses gradient maps as universal features to detect upsampling artifacts.	**BS**: 16 **LR**: 10^{-4} **Input**: Gradient Map → Resize 256 → Random Crop 224 **Aug**: Gaussian Blur (10%), JPEG Compression (10%), and Random Horizontal Flip (50%) **Norm**: ImageNet Stats; Mean: [0.485, 0.456, 0.406], Std: [0.229, 0.224, 0.225]
LNP	Detects forgery via denoising residual maps from local noise estimation.	**BS**: 16 **LR**: 10^{-4} **Input**: LNP Denoising Map → Resize 256 → Random Crop 224 **Aug**: Gaussian Blur (10%), JPEG Compression (10%), and Random Horizontal Flip (50%) **Norm**: ImageNet Stats; Mean: [0.485, 0.456, 0.406], Std: [0.229, 0.224, 0.225]
UnivFD	Trains a linear classifier on features from a frozen CLIP visual encoder.	**BS**: 16 **LR**: 10^{-4} **Input**: Resize 256 → Random Crop 224 **Aug**: Gaussian Blur (10%), JPEG Compression (10%), and Random Horizontal Flip (50%) **Norm**: CLIP Stats; Mean: [0.481, 0.458, 0.408], Std: [0.269, 0.261, 0.276]
AIDE	Fuses noise (DCT) and semantic (CLIP) features via Mixture-of-Experts.	**BS**: 16 **LR**: 10^{-4} **Pipeline**: DCT/SRM-based low-frequency analysis; OpenCLIP semantic embedding; feature concatenation and classification **Norm**: ImageNet Stats; Mean: [0.485, 0.456, 0.406], Std: [0.229, 0.224, 0.225]
FatFormer	Forgery detection using a Vision Transformer backbone with standard protocols.	**BS**: 16 **LR**: 2×10^{-5} **Input**: Resize 256 → Random Crop 224 **Aug**: Gaussian Blur (10%), JPEG Compression (10%), and Random Horizontal Flip (50%) **Norm**: ImageNet Stats; Mean: [0.485, 0.456, 0.406], Std: [0.229, 0.224, 0.225]
NPR	Analyzes correlations between neighboring pixels.	**BS**: 16 **LR**: 2×10^{-4} **Input**: Resize 256 → Random Crop 224 **Aug**: Random Horizontal Flip (50%) **Norm**: ImageNet Stats; Mean: [0.485, 0.456, 0.406], Std: [0.229, 0.224, 0.225]
AF	Learns intrinsic fingerprints via VAE-reconstructed perfectly aligned real/fake pairs.	**BS**: 16 **LR**: 10^{-4} **Aug**: Random Resized Crop (20%), Random ColorJitter (80%), Random Grayscale (20%), Random Cutout (20%), Gaussian Noise (20%), Gaussian Blur (50%), JPEG Compression (50%), Rotation, and Random Horizontal Flip (50%) **Input**: RandomCrop 96×96 **Norm**: ImageNet Stats; Mean: [0.485, 0.456, 0.406], Std: [0.229, 0.224, 0.225]
Effort	Uses SVD-based orthogonal subspace decomposition, freezing principal components to preserve pre-trained knowledge while adapting residual components for forgery detection.	**BS**: 16 **LR**: 2×10^{-4} **Input**: Random Crop 224 **Aug**: Random Horizontal Flip (50%) **Norm**: CLIP Stats; Mean: [0.481, 0.458, 0.408], Std: [0.269, 0.261, 0.276]
PhysCons (Ours)	Synthetic content is identified by detecting violations of intrinsic physical consistency in images.	**BS**: 16 **LR**: 10^{-4} **Input**: Resize 256 → Random Crop 224 **Aug**: Random Horizontal Flip (50%) **Norm**: ImageNet Stats; Mean: [0.485, 0.456, 0.406], Std: [0.229, 0.224, 0.225]

generalization across unseen generators. The full PhysCons framework achieves an average AUC of 0.9821, outperforming the runner-up baseline Fusing (0.9539) and the recent AIDE (0.9077). In addition, the strong standalone performance of individual branches, especially the luminance–

Table 13: **Cross-generator evaluation results on the GenImage dataset.** ROC-AUC scores of different detectors in detecting real and fake images from different generators. These methods are trained on real images from ImageNet and fake images generated by SD v1.4 and evaluated over eight generators. **Bold** and underline indicate the best and second-best results.

Methods	Venue	SD v1.4	ADM	BigGAN	GLIDE	Midjourney	SD v1.5	VQDM	Wukong	Avg.
CNNSpot (Wang et al., 2020)	CVPR'20	0.9999	0.5062	0.8044	0.7991	0.8504	0.9997	0.4198	0.9994	0.7974
FreDect (Frank et al., 2020)	ICML'20	0.9994	0.2521	0.4750	0.6719	0.8785	0.9987	0.2564	0.9970	0.6911
Gram-Net (Liu et al., 2020)	CVPR'20	0.9975	0.4637	0.7227	0.7230	0.6936	0.9971	0.4517	0.9899	0.7549
Fusing (Ju et al., 2022)	ICIP'22	**1.0000**	0.9076	0.8840	0.9676	0.8998	**1.0000**	0.9725	**1.0000**	0.9539
LNP (Liu et al., 2022)	ECCV'22	**1.0000**	0.7630	0.8950	0.7935	0.7841	**1.0000**	0.5799	**1.0000**	0.8519
LGrad (Tan et al., 2023)	CVPR'23	0.9948	0.4805	0.7506	0.7191	0.7640	0.9941	0.6024	0.9826	0.7860
UnivFD (Ojha et al., 2023)	CVPR'23	0.9957	0.6085	0.8766	0.9410	0.9395	0.9958	0.7239	0.9779	0.8824
FatFormer (Liu et al., 2024)	CVPR'24	**1.0000**	0.6608	0.9611	0.9476	0.9267	0.9995	0.7241	0.9998	0.9025
NPR (Tan et al., 2024)	CVPR'24	0.9999	0.4800	0.9200	0.9820	0.9670	0.9990	0.4990	0.9990	0.8557
AF (Sundara Rajan et al., 2025)	ICLR'25	**1.0000**	0.8075	0.7823	0.8608	0.7823	0.9998	0.8936	0.9999	0.8908
AIDE (Yan et al., 2025a)	ICLR'25	**1.0000**	0.8637	0.7136	0.9791	0.8472	0.9975	0.8613	0.9998	0.9077
PhysCons-SS (Ours)	-	0.9977	0.6613	0.9924	0.9614	0.7899	0.9964	0.7187	0.9887	0.8883
PhysCons-LC (Ours)	-	0.9997	**0.9429**	0.9974	**0.9950**	**0.9882**	0.9994	**0.9760**	0.9997	**0.9873**
PhysCons-NT (Ours)	-	0.9985	0.8734	0.9641	0.9110	0.9301	0.9984	0.8849	0.9963	0.9446
PhysCons (Ours)	-	0.9995	0.9382	**0.9987**	0.9939	0.9661	0.9992	0.9630	0.9979	0.9821

Table 14: **Cross-generator evaluation results on the AIGCDetectBenchmark dataset.** ROC-AUC scores of different detectors in detecting real and fake images from different generators. These methods are trained on real images from LSUN and fake images generated by ProGAN and evaluated over sixteen generators. **Bold** and underline indicate the best and second-best results.

Methods	*ProGAN*	*ADM*	*BigGAN*	*CycleGAN*	*DALLE2*	*GauGAN*	*GLIDE*	*Midjourney*	*SD v1.4*	*SD v1.5*	*StarGAN*	*StyleGAN*	*StyleGAN2*	*VQDM*	*WFIR*	*Wukong*	*Avg.*
CNNSpot (Wang et al., 2020)	**1.0000**	0.7217	0.8429	0.9144	0.5010	0.8947	0.7054	0.5756	0.5969	0.6062	0.9279	0.9768	0.9640	0.6507	0.7678	0.5409	0.7617
FreDect (Frank et al., 2020)	**1.0000**	0.6910	0.8922	0.6584	0.1340	0.8473	0.4082	0.2619	0.2058	0.2035	0.9999	0.8620	0.8808	0.8409	0.4233	0.3037	0.6008
Gram-Net (Liu et al., 2020)	0.9999	0.5617	0.7655	0.8621	0.5365	0.7690	0.5035	0.5552	0.5376	0.5372	0.9479	0.9779	0.9944	0.6317	0.7520	0.4860	0.7136
Fusing (Ju et al., 2022)	**1.0000**	0.8986	0.7089	0.9292	**0.9962**	0.5166	0.8645	0.7075	0.9703	0.9702	**1.0000**	0.8905	0.9561	0.9221	0.9401	0.9416	0.8883
LNP (Liu et al., 2022)	**1.0000**	0.8080	0.8906	0.7106	0.9587	0.7743	0.7703	0.7384	0.5863	0.5826	**1.0000**	0.9122	0.9654	0.5858	0.1503	0.6132	0.7529
LGrad (Tan et al., 2023)	0.9999	0.6733	0.8647	0.8704	0.7897	0.8412	0.7385	0.7179	0.6550	0.6599	0.9992	0.9874	0.9842	0.6897	0.6226	0.6097	0.7940
UnivFD (Ojha et al., 2023)	**1.0000**	0.8940	0.9943	**0.9983**	0.6417	**0.9999**	0.8637	0.4756	0.6985	0.6995	0.9947	0.9880	0.9903	0.9732	0.9480	0.7891	0.8718
FatFormer (Liu et al., 2024)	**1.0000**	0.6844	0.9187	0.9722	0.5025	0.8546	0.5916	0.6132	0.7451	0.7487	**1.0000**	0.9847	0.9984	0.6358	0.8019	0.6774	0.7955
NPR (Tan et al., 2024)	**1.0000**	0.7552	0.8832	0.9913	0.8627	0.7945	0.8728	0.9090	0.8992	0.8986	0.9993	0.9956	**0.9996**	0.7228	0.6977	0.8167	0.8811
AF (Sundara Rajan et al., 2025)	**1.0000**	0.6992	**0.9973**	0.9878	0.7171	0.9980	0.8418	0.6710	0.7521	0.7510	0.9999	**1.0000**	0.9994	**0.9994**	**0.9866**	0.7678	0.8855
AIDE (Yan et al., 2025a)	0.9979	0.9199	0.7183	0.9457	0.9469	0.6138	0.9616	0.7872	0.9327	0.9417	**1.0000**	0.9990	0.9963	0.8873	0.9772	0.9432	0.9105
PhysCons-SS (Ours)	0.9961	0.6016	0.9778	0.9180	0.8606	0.9216	0.9763	0.7711	0.8962	0.8929	0.9947	0.8151	0.8299	0.7799	0.5572	0.8790	0.8543
PhysCons-LC (Ours)	0.9997	**0.9310**	0.9446	0.9385	0.9537	0.9987	0.9718	0.9077	0.9651	0.9655	0.9989	0.9930	0.9878	0.9430	0.7126	0.9454	**0.9473**
PhysCons-NT (Ours)	0.9988	0.8923	0.9419	0.9767	0.8730	0.9485	0.9730	0.8743	0.9254	0.9228	**1.0000**	0.9468	0.9499	0.9248	0.5400	0.9030	0.9120
PhysCons (Ours)	0.9996	0.9308	0.9575	0.9470	0.9470	0.9893	**0.9883**	**0.9159**	**0.9720**	**0.9714**	0.9999	0.9753	0.9610	0.9484	0.5747	**0.9546**	0.9395

Table 15: **Evaluation on challenging and realistic scenarios**. ROC-AUC scores of different detectors on Chameleon and Sora. All models are trained exclusively on ProGAN. **Bold** and underline indicate the best and second-best results.

Source	Methods															
	CNNSpot	FreDect	Fusing	LNP	LGrad	Gram-Net	UnivFD	FatFormer	AF	AIDE	NPR	Effort	PhysCons -SS (Ours)	PhysCons -LC (Ours)	PhysCons -NT (Ours)	PhysCons (Ours)
Chameleon	0.3207	0.5592	0.7023	0.6232	0.4745	0.4517	0.3990	0.5075	0.4160	0.5562	0.5828	0.5728	0.7336	0.5839	**0.7926**	0.7507
Sora	0.3066	0.5409	0.6361	0.6137	0.2211	0.4311	0.3115	0.4003	0.6337	0.4927	0.5158	0.5813	0.6458	0.6017	**0.6813**	0.6522
Average	0.3137	0.5501	0.6692	0.6185	0.3478	0.4414	0.3553	0.4539	0.5249	0.5245	0.5493	0.5771	0.6897	0.5928	**0.7370**	0.7015

chrominance branch PhysCons-LC with an average AUC of 0.9873, further supports our hypothesis that modeling physically coupled consistency provides a stable basis for AI-generated image detection.

C.2 ROC-AUC Evaluation on AIGCDetectBenchmark

We further evaluate ROC-AUC on AIGCDetectBenchmark in Table 14. This setting is more challenging because detectors are trained on ProGAN and evaluated on 16 generators spanning different architectures, resolutions, and visual domains. PhysCons generalizes well from GAN-based training data to modern diffusion generators, including DALLE2, SD v1.4, and SD v1.5. Notably, PhysCons-LC achieves the highest average AUC of 0.9473 in this transfer scenario. These results indicate that PhysCons does not merely memorize source-generator artifacts, but instead captures physical inconsistency patterns that remain informative across generator families.

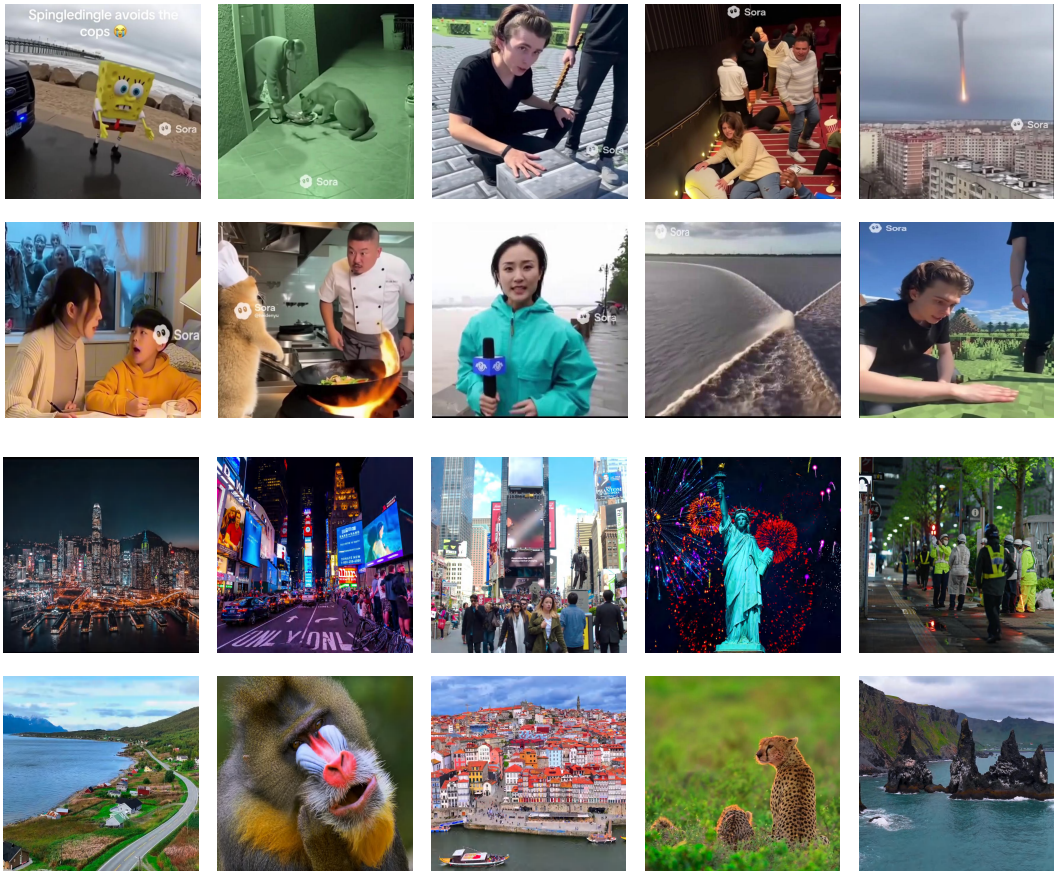


Figure 4: **Visualization of the collected Sora-based dataset.** The top two rows display AI-generated frames produced by Sora, while the bottom two rows consist of real-world frames collected from online platforms. This dataset covers diverse scenarios to rigorously evaluate the generalization capability of detectors against state-of-the-art text-to-video models.

C.3 Evaluation on Challenging and Realistic Scenarios

Table 15 reports the ROC-AUC performance on challenging real-world and state-of-the-art generative scenarios, including Chameleon and Sora. All models are trained on ProGAN and directly evaluated on these unseen datasets. Existing methods often suffer from severe generalization degradation, with some methods approaching near-random performance. In contrast, PhysCons variants maintain stronger robustness. In particular, PhysCons-NT achieves the best AUC on Chameleon (0.7926) and Sora (0.6813), demonstrating that physical consistency modeling remains effective for highly realistic diffusion-generated and text-to-video-generated content.

C.4 Generalization to Recent Generative Models

To further assess whether PhysCons remains effective against recent and advanced generators, we additionally evaluate on images generated by FLUX.1, SD3.5, and gpt-image-2. For each generator, we collect 1,000 images with prompts covering diverse categories, including animals, landscapes, human subjects, and outdoor scenes. All methods are evaluated following the same cross-generator protocol used for GenImage.

As shown in Table 16, PhysCons achieves the best accuracy on FLUX.1, SD3.5 and gpt-image-2. This result is important because these generators are not included in the training data and represent more recent synthesis pipelines. Compared with detectors that rely heavily on generator-specific artifacts, PhysCons exhibits better robustness to newly emerging generative models by detecting violations of physically coupled image statistics.

Table 16: **Generalization to recent high-quality generators.** Accuracy (%) is reported on images generated by FLUX.1, SD3.5, and gpt-image-2.

Generator	NPR	Effort	AF	AIDE	PhysCons
FLUX.1	71.90	64.90	57.10	70.80	**91.60**
SD3.5	56.70	80.50	76.00	67.10	**82.40**
gpt-image-2	56.80	74.30	55.20	68.80	**75.90**

Table 17: **Robustness to image processing and AI manipulation.** Accuracy (%) is reported for HDR-like exposure fusion on real images and AI manipulation via inpainting and super-resolution. The best result in each row is highlighted in bold.

Setting	Type	NPR	Effort	AF	AIDE	PhysCons
HDR-like exposure fusion	processed-real robustness	68.08	82.51	81.32	80.98	**86.93**
Inpainting	AI-manipulated robustness	51.00	64.70	58.70	57.20	**65.60**
Super-resolution	AI-manipulated robustness	67.19	76.82	61.05	71.38	**79.61**

C.5 Robustness to Image Processing and AI Manipulation

We further evaluate detector robustness to image processing and AI manipulation in three settings: HDR-like exposure fusion, inpainting, and super-resolution. HDR-like exposure fusion assesses whether processed real images remain correctly classified as real, while inpainting and super-resolution assess whether AI-manipulated images are correctly identified as fake.

Table 17 reports detection accuracy under image processing and AI manipulation, covering HDR-like exposure fusion on real images and AI manipulation via inpainting and super-resolution. PhysCons achieves the highest accuracy among the compared methods across all three settings, supporting its robustness to the evaluated processing and manipulation operations.

C.6 JPEG Robustness with Compression-Aware Training

Our main experiments intentionally do not include adversarial training or JPEG-compressed images during training, so that the intrinsic generalization ability of PhysCons can be evaluated without additional robustness-oriented augmentation. However, since JPEG compression is common on social media platforms, we further examine whether compression-aware training can improve practical robustness. Specifically, during training, JPEG compression is applied to each image with a probability of 30%, where the quality factor is uniformly sampled from 50 to 100.

Table 18: **JPEG robustness with compression-aware training.** Accuracy of PhysCons under different JPEG quality factors after adding JPEG augmentation during training.

JPEG quality factor	$Q = 85$	$Q = 75$	$Q = 50$
PhysCons w/ JPEG augmentation	0.7955	0.7321	0.6966

As shown in Table 18, JPEG augmentation substantially improves the robustness of PhysCons under compression. Even under aggressive compression with $Q = 50$, the model still achieves 0.6966 accuracy, clearly above random guessing. This suggests that the proposed physical consistency framework can be adapted to practical deployment conditions when compression-aware augmentation is included.

C.7 Effect of Ensemble Strategy

PhysCons adopts a Max-Ensemble strategy to aggregate the predictions of the three physical-consistency branches. This design follows the intuition that an image should be considered suspicious if any branch detects a strong violation of physical consistency. To examine this design choice, we

Table 19: **Effect of ensemble strategy.** We compare the original Max-Ensemble strategy with an alternative Average-Ensemble strategy. All metrics are computed independently on each GenImage subset and then averaged; thus, F1 is not derived from the averaged precision and recall.

Method	Accuracy	Precision	Recall	F1-score
PhysCons-SS	78.99	84.57	61.55	66.18
PhysCons-LC	88.60	**96.71**	79.01	85.51
PhysCons-NT	83.03	96.15	68.00	78.06
PhysCons w/ Average-Ensemble	90.16	95.63	83.47	88.05
PhysCons w/ Max-Ensemble	**92.23**	95.36	**88.42**	**91.27**

Table 20: **Fixed SRM filter vs. learnable BayarConv in the Noise–Texture branch.** Accuracy is reported under the GenImage cross-generator setting.

Method	SD v1.4	ADM	BigGAN	GLIDE	Midjourney	SD v1.5	VQDM	Wukong	Avg.
BayarConv variant	**99.82**	63.34	77.36	**81.63**	**71.49**	**98.99**	68.47	**97.94**	82.38
Original SRM	98.23	**66.59**	**84.57**	79.47	70.33	97.94	**72.50**	94.58	**83.03**

further compare Max-Ensemble with an alternative Average-Ensemble strategy, which averages the prediction probabilities of all branches before making the final decision.

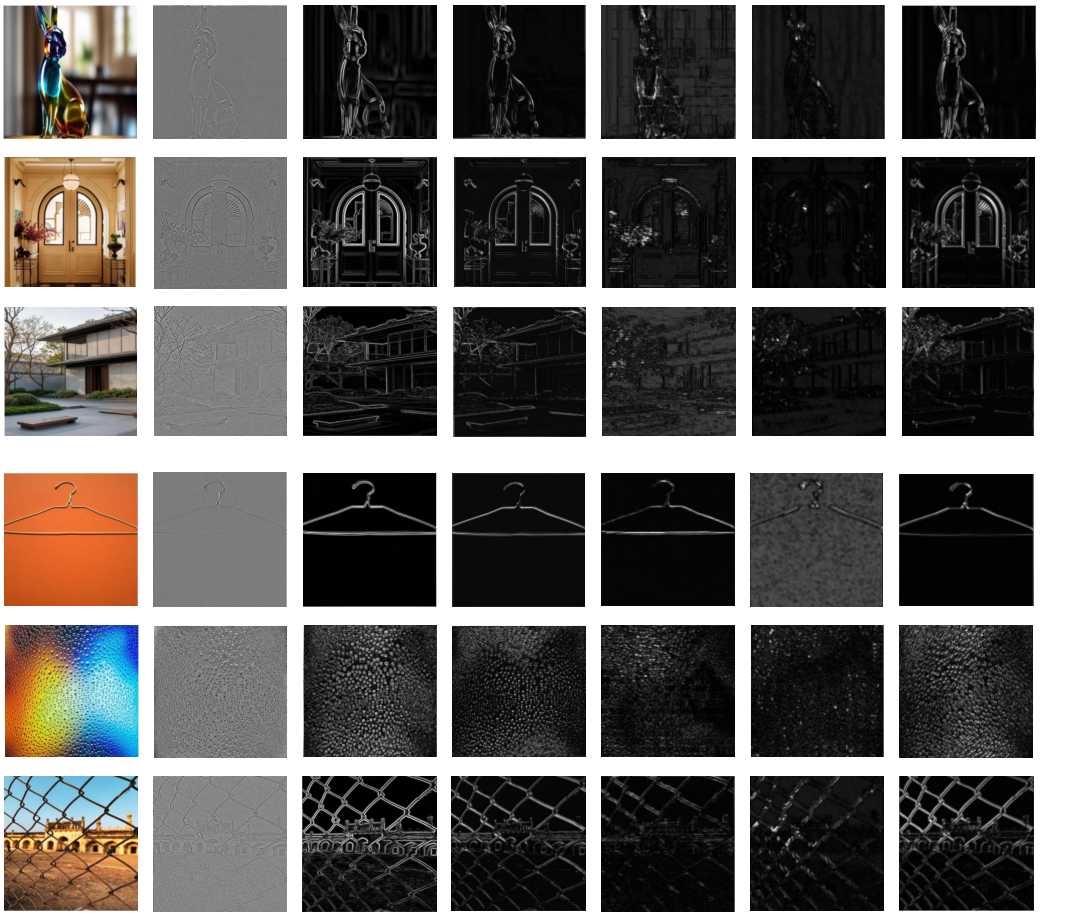
Table 19 compares different strategies for aggregating the three physical-consistency branches. Average-Ensemble already outperforms all individual branches in terms of Accuracy and F1-score, indicating that the proposed branches provide complementary cues. Nevertheless, averaging may smooth out sparse but highly discriminative inconsistency responses from a single branch. By contrast, Max-Ensemble explicitly preserves the strongest physical-inconsistency evidence among the branches. Compared with Average-Ensemble, Max-Ensemble improves accuracy, recall, and F1 score by 2.07, 4.95, and 3.22 percentage points. These results support the use of Max-Ensemble for its higher accuracy, recall, and F1 score under the evaluated setting.

The purpose of integrating multiple physical dimensions is to retain detections arising from different types of inconsistency. With identical branch outputs and a shared threshold, max aggregation preserves every detection made by any constituent branch. This preserves recall while allowing additional false positives, explaining why the ensemble need not achieve the highest accuracy on every benchmark. The benefit of combining cues is demonstrated on GenImage, where the full ensemble achieves 92.23% accuracy compared with 88.60% for the strongest individual branch, together with the highest recall of 88.42% among the evaluated configurations (Table 7). A practical application of this design is efficient pre-screening in large-scale content moderation and forensic workflows, where flagged images undergo subsequent human review and verification using additional evidence. With only 0.05M parameters and a throughput of 722 FPS on an NVIDIA RTX 4090, the full PhysCons ensemble combines inexpensive screening with Physical Violation Maps that provide spatial cues for further inspection.

C.8 Fixed Physical Operators vs. Learnable Alternatives

PhysCons uses fixed physical operators such as Sobel and SRM filters to explicitly encode prior knowledge about image formation. To examine whether these operators can be replaced by learnable alternatives, we conduct an experiment on the Noise–Texture branch by replacing the fixed SRM filter with a learnable constrained convolution, i.e., BayarConv. The variant is trained on Stable Diffusion v1.4 and evaluated on unseen generators.

The learnable BayarConv variant improves performance on several generators, such as SD v1.4, GLIDE, Midjourney, SD v1.5, and Wukong. However, the original SRM-based design achieves higher average accuracy. This indicates that learnable residual extraction may increase flexibility on certain domains, but fixed physical priors can provide stronger cross-generator stability. Therefore, we retain the fixed SRM filter in the main framework.



(a) Input (b) Spectral (c) Spatial (d) Luminance (e) Chrominance (f) Noise Res. (g) Texture Comp.

Figure 5: **Visualization of the three physically coupled feature pairs used in PhysCons.** The top three rows display **AI-generated images**, while the bottom three rows show **real images**. The columns correspond to the three physical dimensions defined in our framework: (a) Original RGB Image; **Dimension I (Spectral–Spatial):** (b) Spectral Phase Structure and (c) Spatial Gradient Magnitude; **Dimension II (Luminance–Chrominance):** (d) Luminance Map and (e) Chrominance Map; **Dimension III (Noise–Texture):** (f) Noise Residual Map and (g) Texture Complexity Map.

D Visualization of Physical Inconsistency Discovery

D.1 Visualization of Physical Inconsistency Discovery

To intuitively demonstrate the effectiveness of our proposed PhysCons framework, we provide a detailed visualization of the three physically coupled feature pairs defined in our methodology. Figure 5 presents a side-by-side comparison between AI-generated images (top three rows) and real images (bottom three rows) across seven columns. These columns correspond to the original input and the three physical dimensions: **Dimension I (Spectral–Spatial)**, **Dimension II (Luminance–Chrominance)**, and **Dimension III (Noise–Texture)**.

Analysis of Real Images (Bottom Rows): As observed in the bottom three rows (e.g., the hanger, the water droplets, and the wire fence), real images exhibit strong intrinsic physical consistency. Specifically:

- **Spectral–Spatial Consistency (Cols b vs. c):** The spectral phase structure reconstructed via IFFT (Col b) aligns perfectly with the spatial gradient magnitude extracted via Sobel filters (Col c). The edges are sharp, continuous, and spatially synchronized.
- **Luminance–chrominance Coupling (Cols d vs. e):** The luminance map (Col d) and chrominance map (Col e) share identical structural boundaries. For instance, the intricate details of the wire fence are preserved consistently across both light and color channels without bleeding or inconsistency.
- **Noise–Texture Dependency (Cols f vs. g):** The noise residuals (Col f) are highly correlated with the texture complexity map (Col g). High-frequency noise is naturally masked by complex textures, adhering to the signal-dependent noise model of camera ISPs.

Analysis of AI-Generated Images (Top Rows): In stark contrast, the AI-generated images in the top three rows (e.g., the glass rabbit, the doorway, and the building) display visible **physical inconsistencies** or "decoupling" artifacts:

- **Structural Hallucinations:** In the first row (glass rabbit), the chrominance map (Col e) exhibits unnatural artifacts that do not exist in the luminance map (Col d), violating the luminance-chrominance coupling.
- **Noise Mismatch:** In the second row (doorway), the noise residual map (Col f) shows a scattered, uniform pattern that fails to adapt to the local texture complexity shown in Col g. This suggests the generator acts as a global filter rather than simulating physical sensor noise.
- **Spectral Artifacts:** In the third row (building), the spectral phase structure (Col b) contains high-frequency artifacts in the sky region, which are absent in the spatial gradient map (Col c).

These visualizations empirically validate our core hypothesis: while generative models can produce perceptually realistic RGB images (Col a), they struggle to maintain the rigorous physical constraints inherent in the image formation process. PhysCons effectively leverages these subtle yet pervasive inconsistencies for robust detection.

D.2 Interpretability Analysis of Physical Consistency Branches

To validate that our proposed PhysCons framework truly learns distinct physical cues rather than relying on spurious correlations, we visualize the violation maps of the three individual physical branches and the Grad-CAM heatmaps of three state-of-the-art methods in Fig. 6. The rightmost panel of Fig. 6 explicitly breaks down the focus of each branch, confirming our design motivation:

- **Spectral–Spatial Branch:** As observed in the top right panel, the attention maps of this branch are sharply concentrated on high-frequency structural boundaries, such as hair strands, facial contours, and clothing folds. This empirically verifies that the module effectively detects the inconsistency between spectral phase information and spatial gradients, which typically manifests at sharp edges where generative models often produce "ringing" or "blurring" artifacts.
- **Luminance–Chrominance Branch:** The middle right panel shows that this branch predominantly activates in regions with strong lighting transitions and shadows, such as the shading on faces, neck areas, and bright reflections. This behavior aligns with the physical principle that luminance and chrominance should be coupled during natural image formation. The model successfully identifies "unnatural shadows" or color bleeding at light-dark boundaries, which are common weaknesses in current image generators.
- **Noise–Texture Branch:** In the bottom right panel, the violation maps exhibit a distinct block-wise or regional activation pattern. The model focuses on areas with specific texture complexity, such as dense animal fur, intricate fabric patterns, or background foliage. This indicates that the branch is assessing the statistical dependency between local texture content and noise residuals, flagging regions where the synthesized texture lacks the consistent sensor noise characteristics found in real images.



(a) NPR (b) FatFormer (c) AIDE (d) PhysCons

Figure 6: **Visualization of Grad-CAM heatmaps and violation maps.** We compare the feature activations of our proposed **PhysCons** (d) against state-of-the-art methods: (a) NPR, (b) FatFormer, and (c) AIDE. While existing methods often exhibit scattered noise or sparse focus, PhysCons consistently targets comprehensive forgery regions. The rightmost panel highlights the interpretability of our method by visualizing the specific activations from three physically-grounded branches: *Spectral-Spatial*, *Luminance-Chrominance*, and *Noise-Texture*, demonstrating the complementary nature of these cues in detection.

In contrast to comparison methods (left columns) like NPR (which shows scattered, noisy activations) or AIDE (which focuses broadly on semantic objects), PhysCons provides precise, physically interpretable localization of forgery traces. This disentangled feature learning strategy is key to our method's robust generalization capability.

D.3 Feature Space Visualization and Generalization Analysis

To qualitatively evaluate the generalization capability of the learned representations, we visualize the feature distributions of different detection methods using t-SNE. Fig. 7 compares the feature manifolds of five baselines (left panel) against the three physical branches of our PhysCons framework (right panel). The visualization covers two distinct scenarios: an *in-domain* setting (Seen Generator: SD v1.4) and a challenging *out-of-domain* setting (Chameleon).

Baselines Analysis (Left Panel): As illustrated in the left columns, existing methods such as Effort, FatFormer, and NPR form distinct clusters for real and fake images on the seen generator (SD v1.4). However, their performance collapses on the unseen Chameleon dataset. The feature distributions of real (blue) and fake (red) images exhibit severe overlapping and entanglement, indicating that these models have overfitted to generator-specific artifacts (e.g., specific noise patterns of Stable Diffusion) that do not transfer to other domains.

PhysCons Analysis (Right Panel): In stark contrast, the three branches of PhysCons (SS, LC, and NT) demonstrate remarkable robustness.

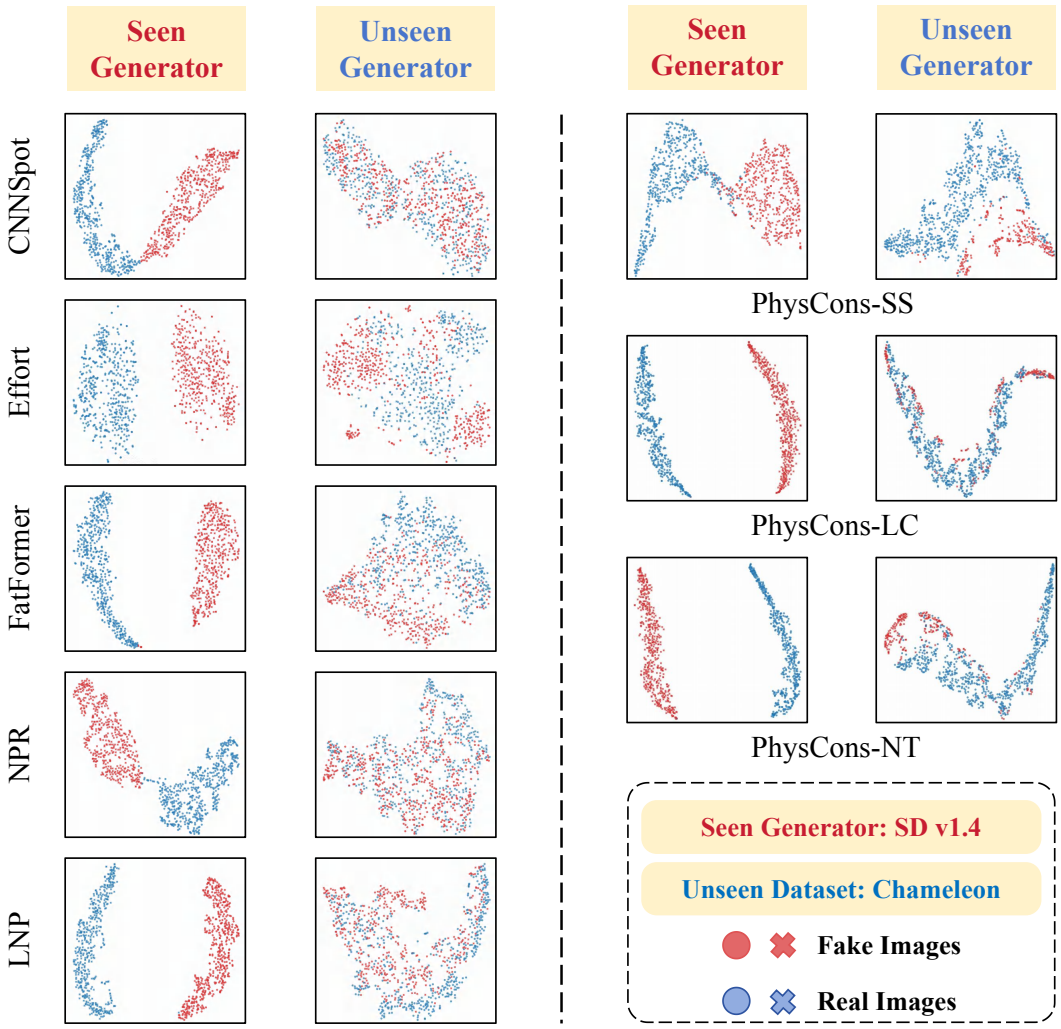


Figure 7: **t-SNE visualization of feature embeddings.** All models are trained exclusively on the SD v1.4 subset of GenImage. We visualize the feature distributions on the in-domain SD v1.4 test set and the cross-domain Chameleon dataset.

- **In-Domain:** On SD v1.4, our methods achieve clear separation similar to or better than the baselines.
- **Out-of-Domain:** Crucially, on the unseen Chameleon dataset, our physical features still maintain relatively well-separated cluster structures. For instance, the PhysCons-SS branch exhibits a particularly sharp decision boundary, effectively distinguishing diverse unknown forgeries from real images.

This visualization provides strong empirical evidence for our core hypothesis: while statistical artifacts vary wildly across generators, the violations of intrinsic physical consistencies (e.g., lighting, texture, and spectral structure) remain a universal fingerprint of AI synthesis. PhysCons successfully captures these domain-invariant physical cues, enabling robust generalization where other methods fail.