A Debiased Reconstruction-based Framework for Training-Free Detection of AI-Generated Images

Sungik Choi
LG AI Research
sungik.choi@lgresearch.ai

Hankook Lee
Sungkyunkwan University

Jaehoon Lee
LG AI Research

Robin Kim
University of Massachusetts at Amherst

Stanley Jungkyu Choi
LG AI Research

Moontae Lee
LG AI Research

Abstract

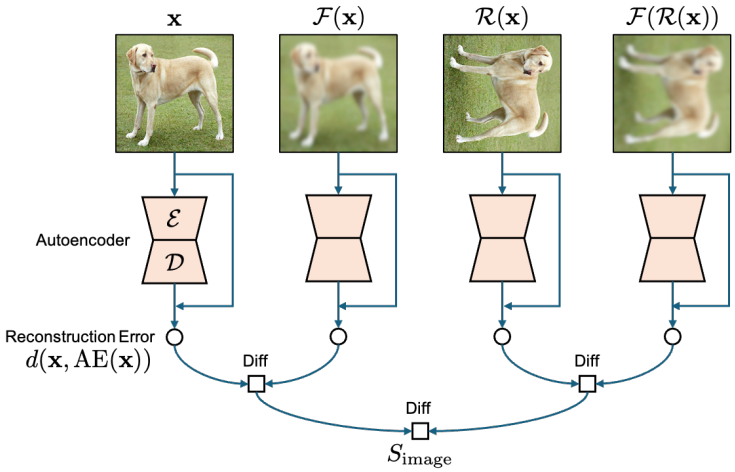
As recent AI models have successfully generated high-resolution photorealistic images, it has also been socially important to detect whether an image is generated by AI. Since training data for the detection task is often not available due to the diversity of generative models, training-free detection approaches have been practically considered. A common approach is to utilize the image-level reconstruction error from the latent diffusion model (LDM). However, we find this score suffers from instance-specific biases, particularly in images with simple backgrounds. To this end, we propose a novel image-level debiasing score function that cancels out background contribution by normalizing the reconstruction error on the augmented images with similar background information. To be specific, we show that rotation and low-pass filtering are effective augmentation strategies. To promote generalization to broader generative models, we newly explore latent-level reconstruction error as an additional training-free signal. However, we observe that the latent-level score also suffers to latent-specific bias. To mitigate this, we introduce a rotation-based latent-level debiasing score based on the normalization of the rotated latent. We unify the aforementioned scores into a single unified debiasing score, RDD, which achieves state-of-the-art training-free detection performance across diverse generative models. Furthermore, our framework can be robust to corruption of the examined images.

1. Introduction

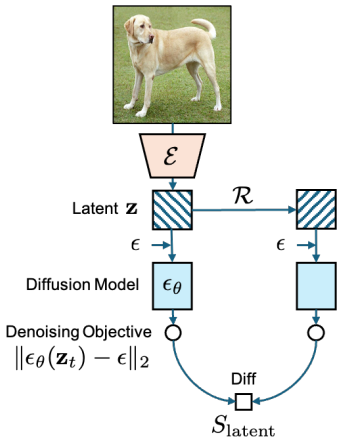
Recent advances in text-to-image (T2I) generative models have enabled the creation of highly realistic images that are often indistinguishable from real ones. While these generative capabilities are impressive, they have also led to several negative societal impacts, such as causing societal confusion [28] or infringing upon intellectual property rights [1]. To mitigate such misuse, the task of detecting AI-generated images (AIGIs) has become increasingly important. Most existing AIGI detection methods assume access to the distribution of real and generated images during training, allowing them to learn discriminative features that differentiate real and generated images. However, this assumption may not hold in practical scenarios due to the wide variety and rapid evolution of T2I generative models. Motivated by these challenges, we study the *training-free* detection scenario, where no real or AI-generated images are available during training a detector.

Existing training-free methods for detecting AIGIs typically leverage pre-trained foundation models to design detection score functions without requiring labeled real or synthetic data. A common approach relies on *image-level* reconstruction error [19] using the autoencoder component of a latent diffusion model (LDM) [20]. While appealing for their simplicity and training-free nature, these methods suffer from inherent limitations. Notably, they tend to underestimate reconstruction error for real images with simple textures, leading to false positives. Moreover, their reliance on the autoencoder makes the approach fail to generalize well to AIGIs generated by other types of models (e.g., GANs).

In this paper, we identify a key limitation of existing approaches, *background bias*, where images with simple textures or backgrounds yield low image-level reconstruction error, irrespective of whether they are real or AI-generated. To this end, we introduce a novel test-time debiasing framework based on data augmentation. Specifically, we design augmentations that retain background information while inducing increased reconstruction error primarily in AIGIs. By normalizing the original reconstruction error with that of the augmented image, we effectively suppress background-related spurious correlations. We identify that low-pass fil-



(a) Overview of our image-level debiasing score function.



(b) Overview of our latent-level debiasing score function.

Figure 1. Overview of our debiasing framework, RDD. RDD utilizes a joint estimation of image-level and latent-level denoising objective for broader detection of AI-generated images.

tering and rotation serve as effective augmentations for constructing our debiased score, shown in Figure 1a.

To enhance generalization across a wider spectrum of AIGIs, we extend our approach to the latent space of the diffusion model. Although *latent-level* reconstruction error alone shows limited effectiveness, we propose a rotation-based latent debiasing score to improve detection. This calibrated score, visualized in Figure 1b, significantly improves detection of visually discrepant AIGIs.

We further integrate our image-level and latent-level debiased scores into a unified detection metric, termed the Reconstruction Debiasing Detection (RDD) score. RDD consistently outperforms existing training-free detectors across a range of benchmarks, including LDMs, GANs, and image-level diffusion models. RDD also demonstrates strong robustness to test-time perturbations. Lastly, we show that our image-level debiased score enables efficient and accurate attribution of LDM-generated images.

In summary, our contributions are as follows:

- We identify a background bias in the image-level reconstruction error and propose an image-level debiasing score function.
- We also explore latent-level reconstruction error and propose our debiased reconstruction error function. To our knowledge, this is the first practice to utilize latent-level reconstruction error for training-free AIGI detection.
- We evaluate our unified framework across a wide range of generative models and show consistent performance improvements.
- Our framework can also be applied to trace the source of LDM-generated images.

2. Preliminaries

2.1. AI-generated image detection

The goal of this paper is to distinguish between AI-generated and real images. Formally, given the real data distribution $p_{\text{real}}(\mathbf{x})$ and model distribution $p_G(\mathbf{x})$ where G is a generative model trained to mimic p_{real}, we aim to design a score function $S(\mathbf{x})$ that determines whether $\mathbf{x}$ is from $p_{\text{real}}(\mathbf{x})$ (*i.e.*, $S(\mathbf{x}) > \tau$) or not (*i.e.*, $S(\mathbf{x}) \leq \tau$). To consider practical scenarios, we further assume that the generative model G is unknown and the real data $\mathbf{x} \sim p_{\text{real}}(\mathbf{x})$ used for training G is not available. Namely, $S(\mathbf{x})$ is designed to be a universal metric without prior knowledge of generative models. We refer to this setup as "training-free" AI-generated image detection and its counterpart as "training-based".

Only a few works discuss a setting where the inspected real data is unknown. RIGID [6] and MINDER [23] utilize a perturbation sensitivity of self-supervised models on simple augmentations like Gaussian blurring or Gaussian Noise. ZED [4] proposes training an image compressor on COCO images and applying its representation to other images. However, it may not generalize to the resized or noised images. AEROBLADE [19] applies AE reconstruction error of LDMs. Finally, Manifold induced bias [2] measures the similarity of the image and the predicted noise of the latent space in the CLIP [16] embedding space, which is combined with other metrics. However, to our knowledge, no methods have considered the joint utilization of pixel-level and latent-level detection scores.

Several works [12, 24] have discussed utilizing the latent representation of the LDM. However, the key idea has been training a classifier on the extracted representation. In contrast, we propose a training-free detection method without

Figure 2. **Example of background-removed real images.** We collect images from the class "Jack-o'-lantern" in the ImageNet dataset **(up)**. Background-removed images via CLIPSeg **(down)**.

any training data.

2.2. Latent Diffusion Models

LDM consists of two modules: an autoencoder (AE) and a diffusion model. The autoencoder enables efficient high-resolution generation by modeling the generative process on the latent space $\mathcal{Z} \subset \mathbb{R}^{C' \times H' \times W'}$ instead of the data space $\mathcal{X}$. This latent space is typically modelled by pre-training the encoder $\mathcal{E} : \mathcal{X} \to \mathcal{Z}$ and the decoder $\mathcal{D} : \mathcal{Z} \to \mathcal{X}$. While the VAE [8] is usually, the latent distribution typically shows negligible variance [20]. Hence, we set $\mathrm{AE}(\mathbf{x}) \coloneqq \mathcal{D} \circ \mathcal{E}(\mathbf{x})$ for the rest of the paper.

Given the trained encoder $\mathcal{E}$, a diffusion model is trained on the given latent projection of real images (*e.g.*, LAION [21]). That is, given the distribution of latent $\mathcal{D}_{\text{latent}}$, a noise prediction module $\epsilon_\theta(\mathbf{z}_t, t, c)$ is trained via the following diffusion objective:

$$L(\theta) = \mathbb{E}_{(\mathbf{z}_0,\mathbf{c})\sim\mathcal{D}_{\text{latent}},t\sim\mathcal{U}\{0,1\},\epsilon\sim\mathcal{N}(\mathbf{0},\mathbf{I})} \|\epsilon_\theta(\mathbf{z}_t, t, \mathbf{c}) - \epsilon\|_2^2,$$

where $\mathbf{z}_t = \sqrt{\bar{\alpha}_t}\mathbf{z}_0 + \sqrt{1-\bar{\alpha}_t}\epsilon$ denotes the forward diffusion process. This noise prediction module is applied consecutively to generate the denoised latent, which is finally applied through the decoder to generate the image.

3. The Proposed Framework

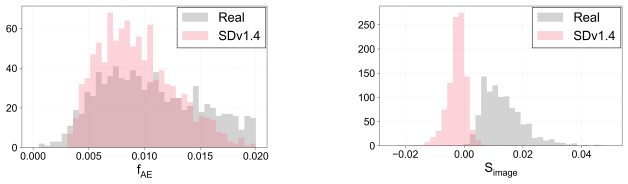
In this section, we present our unified debiasing framework. We begin by identifying instance-specific biases in reconstruction-based detection scores that are unrelated to the forensic distinction between real and AI-generated images. To address this, we introduce a targeted data transformation that preserves these confounding factors while selectively perturbing the features critical for detection. By normalizing reconstruction errors with respect to the transformed inputs, we enhance detection performance at both the image and latent levels. Finally, we propose a unified score function that integrates the debiased score functions without introducing additional hyperparameters.

3.1. A general formulation of debiasing

Given a detection score function $f(\mathbf{x})$ for $\mathbf{x} \in \mathcal{X}$ or $f(\mathbf{z})$ for $\mathbf{z} \in \mathcal{Z}$, assume we have found a transformation technique $\mathcal{T}$ that maps the latent or image data that preserves the bias factor but distorts the forensics information. We propose a debiasing score function S_f for function $f(\mathbf{x})$ as follows:

$$S_{f,\mathcal{T},\lambda}(\mathbf{x}) = f(\mathbf{x}) - \lambda f(\mathcal{T}(\mathbf{x})), \tag{1}$$

where $\lambda \in [0, 1]$ is a transformation-specific hyperparameter. The goal of the debiased score function is twofold: (1) by preserving the bias factor on the $\mathcal{T}(\mathbf{x})$, it should reduce the effect of the bias factor, and (2) the forensic-dependent information will be distorted in the transformed data. In the following sections, we propose such transformations on the image and latent-level reconstruction error, respectively.



(a) (b)

Figure 3. **(a)**: Histogram of f_{AE} on background-removed real and SDv1.4-generated images. **(b)**: Histogram of S_{image} on the same images. SDv1.4 is applied as the autoencoder.

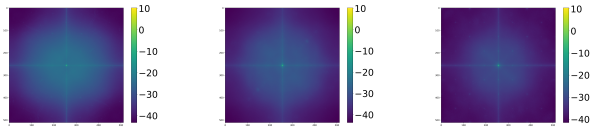
3.2. Image-level debiasing

Baseline The image-level reconstruction error from an autoencoder, defined as $f_{\text{AE}}(\mathbf{x}) = d(\mathbf{x}, \mathrm{AE}(\mathbf{x}))$, where d is a perceptual distance metric (e.g., LPIPS [31]), has been widely used as a detection score [19].

Motivation Ricker et al. [19] observed that the encoder $\mathcal{E}$ tends to map AIGIs onto the latent manifold without significant information loss, enabling near-perfect reconstructions by the decoder $\mathcal{D}$. In contrast, real images, which typically lie off the training distribution, suffer from greater information loss and thus yield higher reconstruction errors. However, we find that f_{AE} is often biased toward instance-specific attributes (e.g., background simplicity) , leading to diminished detection performance. Specifically, the reconstruction error can be underestimated for images with minimal backgrounds. To validate this observation, we design a toy experiment using a dataset comprising 1100 real images of the class "Jack-o'-lantern" from ImageNet [5] and 1100 AI-generated counterparts produced by the SDv1.4 model [20] using the prompt "A photo of a Jack-o'-lantern."

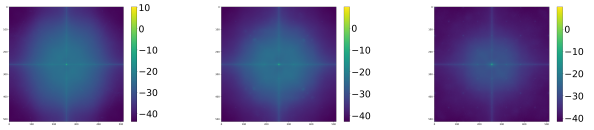
To simulate background bias, we modify the real images by replacing their backgrounds with black using CLIPSeg [11], a zero-shot segmentation model. Background regions are defined as those with low logits in the segmentation mask produced by CLIPSeg under the same text prompt. Figure 2 shows examples before and after segmentation, where semantics are preserved.

As shown in Figure 3a, even when evaluated with the autoencoder from SDv1.4, real images with simplified back-



(a) ImageNet (b) SDv1.4 (c) Midjourney

Figure 4. **Mean power spectra of the difference in the Fourier domain for the $(\mathbf{x}, \mathbf{AE}(\mathbf{x}))$ in real and AIGI data.**



(a) ImageNet (b) SDv1.4 (c) Midjourney

Figure 5. **Mean power spectra of the difference in the Fourier domain for $(\mathbf{R}(\mathbf{x}), \mathbf{AE}(\mathbf{R}(\mathbf{x})))$ in real and AIGI data.**

grounds exhibit reconstruction errors comparable to those of generated images. This demonstrates that f_{AE} fails to effectively distinguish generated content due to its sensitivity to background simplicity, a phenomenon we refer to as *background bias*. Furthermore, even unaltered real images occasionally possess simple backgrounds, exacerbating the degradation in overall detection performance.

Debiased Scores We now propose the transformation for debiasing at the image level. Our two candidates are low-pass filtering and rotation, which preserve the contribution of the background information to the reconstruction error. Moreover, we show that these transforms show higher reconstruction error on the transformed image than the original image when the given image is AI-generated.

First, we show that high-frequency information of the image is informative for the detection of real and AIGIs. We provide the supporting evidence by computing the difference of the given image $\mathbf{x}$ and its reconstruction $\text{AE}(\mathbf{x})$ in the Fourier domain. Figure 4 illustrates the mean power spectra of this difference, computed on the real ImageNet [5], SDv1.4-generated [20], and Midjourney-generated [13] datasets from the GenImage [32] benchmark. Our analysis reveals that real data exhibits greater deviation in high-frequency bands compared to AIGIs.

The aforementioned observations suggest that reconstruction in the real data is easier when high-frequency information is filtered out. Furthermore, as background information is often characterized as low-frequency information [30], applying low-pass filtering would not change the background much. These findings suggest low-pass filtering as an effective transformation method. We formalize our low-pass-filtering-based image debiasing (LFID) score function as follows:

$$S_{\text{LFID}}(\mathbf{x}) = S_{f_{\text{AE}},\mathcal{F},\lambda_{\mathcal{F}}}(\mathbf{x}) = f_{\text{AE}}(\mathbf{x}) - f_{\text{AE}}(\mathcal{F}(\mathbf{x})), \quad (2)$$

where $\mathcal{F} : \mathcal{X} \to \mathcal{X}$ is a low-pass filter. We fix $\lambda_F = 1$ throughout the paper consistently.

We further propose rotation as a debiasing strategy. To be specific, we show that the difference between the rotated image and its reconstruction through the AE is more distinguished in AIGIs than in real images. We verify our hypothesis by comparing the difference between $\mathcal{R}(\mathbf{x})$ and $\text{AE}(\mathcal{R}(\mathbf{x}))$ in the Fourier domain. The results are presented in Figure 5. In comparison to Figure 4, we observe more pronounced distortions in the reconstruction of the rotated data when the data is generated compared to the real data.

Motivated by the observations, we propose the rotation-based image debiasing (RID) score as follows:

$$S_{\text{RID}}(\mathbf{x}) = S_{f_{\text{AE}},\mathcal{R},\lambda_{\mathcal{R}}}(\mathbf{x}) = f_{\text{AE}}(\mathbf{x}) - \lambda_{\mathcal{R}} f_{\text{AE}}(\mathcal{R}(\mathbf{x})), \quad (3)$$

where $\mathcal{R}$ is rotation operation of 90 degrees. Our hyperparameter $\lambda_{\mathcal{R}} \in [0, 1]$ is fixed throughout the experiment.

Since LFID and RID are based on independent data augmentations, unifying them may synergize for refined AI-generated image detection. Our final image-level unified score function S_{image} is defined as follows.

$$\begin{aligned} S_{\text{image}}(\mathbf{x}) &= S_{S_{\text{RID}},\mathcal{F},1}(\mathbf{x}) \\ &= S_{\text{RID}}(\mathbf{x}) - S_{\text{RID}}(\mathcal{F}(\mathbf{x})) \\ &= f_{\text{AE}}(\mathbf{x}) - f_{\text{AE}}(\mathcal{F}(\mathbf{x})) - \lambda_R f_{\text{AE}}(\mathcal{R}(\mathbf{x})) + \lambda_R f_{\text{AE}}(\mathcal{F}(\mathcal{R}(\mathbf{x}))). \end{aligned} \quad (4)$$

S_{image} is defined through the recursive application of low-pass filtering and rotation. Notably, S_{image} is commutative to the order of operations, *i.e.*, $S_{\text{image}}(\mathbf{x}) = S_{\text{LFID}}(\mathbf{x}) - \lambda_R S_{\text{LFID}}(\mathcal{R}(\mathbf{x}))$. We now examine S_{image} on detecting real images with all-black backgrounds introduced in Figure 2. We show the results in Figure 3b, where the S_{image} can successfully detect real images with simple backgrounds.

3.3. Latent-level debiasing

Baseline While the image-level reconstruction error can effectively detect LDM-generated images, it may underperform in detecting non-AE-based models. For example, Figure 6a shows the f_{AE} score experimented in 10000 LSUN images and 10000 images generated by the ProGAN [7] model. f_{AE} fails to distinguish between two distribution.

To extend the generalizability of the detection to the broader generative model, we explore the efficacy of the reconstruction error in the latent space. For a given latent $\mathbf{z}_0$, the latent-level reconstruction error is proposed as follows:

$$\begin{aligned} f_{\text{latent},t}(\mathbf{z}_0) &= \mathbb{E}_{\epsilon\sim\mathcal{N}(\mathbf{0},\mathbf{I})} d_{\text{latent}}(\epsilon_\theta(\mathbf{z}_t, t, \phi), \epsilon) \\ &:= \frac{1}{n}\sum_{i=1}^{n} d_{\text{latent}}(\epsilon_\theta(\mathbf{z}_{t,i}, t, \phi), \epsilon_i), \end{aligned} \quad (5)$$

where d_{latent}, ϕ, $t \in [0, 1]$, and $n \in \mathbb{N}$ denote the latent-level distance function, null-text prompt, diffusion timestep, and

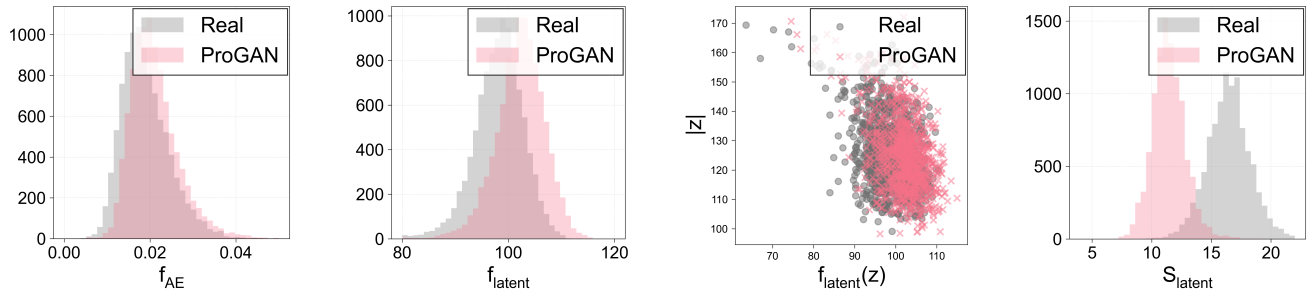


Figure 6. **Motivation of latent debiasing.** **(a)**: Histogram of real and ProGAN-generated data examined by autoencoder reconstruction error. **(b)**: Histogram of real and ProGAN-generated data examined by latent reconstruction error. **(c)**: real and ProGAN-generated data with respect to their latent reconstruction error and latent norm. **(d)**: Histogram of real and ProGAN-generated data examined on our S_{latent}.

the number of noise samples, respectively. t is chosen small enough to guarantee reconstruction of the original latent. The idea of our latent reconstruction error for AIGI detection is based on the observation that real images are applied for $\mathcal{D}_{\text{latent}}$, hence they are likely to be on the manifold of the diffusion model. On the other hand, we expect the generated data not from the given LDM will lie outside the manifold. As our latent-level score $f_{\text{latent}}(z)$ examines the training loss of the diffusion model, we expect our score to be low in real images and high on such off-manifold AIGIs.

We examine our score by testing the same real and AIGIs on the SD-2-base LDM, with d_{latent} as ℓ_2 distance, $n = 16$ and $t = 0.025$, respectively. We show the results in Figure 6b. While the proposed score is more effective than f_{AE}, the discrimination is far from perfect.

Motivation In Figure 3a, we observe several AIGIs assign lower reconstruction error than the real images. To understand this phenomenon, we identify a latent-norm bias, where latents with larger ℓ_2 norm tend to show lower reconstruction error. For verification, we show each sample's latent norm and the latent reconstruction error in Figure 6c. When compared in similar latent norms, the reconstruction error is lower in the real images as expected. However, regardless of whether real or AI-generated, some latents with higher norms tend to assign lower reconstruction error, which deteriorates the overall detection performance.

Debiased Scores We tackle this latent-specific bias by proposing a rotation-based latent-level debiasing score function S_{latent}. S_{latent} normalize the reconstruction error against the rotated latent as follows:

$$\begin{aligned} &S_{\text{latent}}(\mathbf{z}_0) = S_{-f_{\text{latent}},\mathcal{R},1}(\mathbf{z}_0) \\ &= f_{\text{latent},t}(\mathcal{R}(\mathbf{z}_0)) - f_{\text{latent},t}(\mathbf{z}_0) \qquad (6) \\ &\coloneqq \frac{1}{n}\sum_{i=1}^{n} d_{\text{latent}}(\epsilon_\theta(z_{t,i}, t, \phi), \epsilon_i) - d_{\text{latent}}(\epsilon(\mathcal{R}(z)_{t,i}, t, \phi), \epsilon_i). \end{aligned}$$

Our choice of the rotated latent is influenced by S_{RID}, where the rotated image is applied as the OOD data. By normalizing the reconstruction score on the off-manifold latent with the same norm, we expect our S_{latent} to improve on detecting off-manifold AIGIs. We again validate our hypothesis on the same datasets without changing any hyperparameters. We show our results in Figure 6d. Note that S_{latent} greatly improves detection compared to Figure 6b.

3.4. Overall Score

Our proposed reconstruction-based score functions S_{image} and S_{latent} can be unified naturally. To be specific, we define our unified reconstruction-based debiasing detection (RDD) score function as follows:

$$S_{\text{RDD}}(\mathbf{x}) = S_{\text{image}}(\mathbf{x}) \times S_{\text{latent}}(\mathcal{E}(\mathbf{x}))^2. \qquad (7)$$

We apply squared power to S_{latent} since LPIPS distance in S_{image} is the sum of squared ℓ_2 distance in the feature space of the VGG [22] network, analogous to the latent space of the LDM. The multiplication between two scores is chosen since the two scores operate on different ranges. We further compare against other aggregation strategies in the following section.

4. Experiment

This section evaluates the efficacy of RDD in detecting images generated by various generative models. First, we introduce our experiment protocols, benchmarks, and baseline detection methods. We then present our main results, followed by ablation studies and robustness analysis. Finally, we discuss a potential application of our framework in tracing LDM-generated images.

4.1. Experimental Setups

Datasets We evaluate our framework in two widely applied benchmarks that all differ in types of examined generative models, real data distribution, and image resolution. First, GenImage [32] mainly evaluates on latent-diffusion-based T2I generative models against real ImageNet [5].

Table 1. **AI-generated image detection performance (AUROC) of RDD and baselines in the GenImage [32] benchmark. Bold** and underline denote the best and second best methods, respectively.

Method	ADM	BigGAN	GLIDE	Midjourney	SD1.4	SD1.5	VQDM	Wukong	Mean
Training-based Methods									
FatFormer	0.903	0.995	0.951	0.579	0.780	0.776	0.967	0.824	0.847
AIDE	0.921	0.920	0.987	0.959	1.000	1.000	0.965	1.000	0.969
Training-free Methods									
RIGID	0.874	0.974	0.952	0.778	0.682	0.682	0.915	0.699	0.820
MINDER	0.768	0.681	0.582	0.450	0.607	0.596	0.882	0.676	0.655
AEROBLADE	0.851	0.978	0.989	**0.985**	0.976	0.978	0.721	0.978	0.932
Manifold Bias	0.681	0.927	0.822	0.476	0.663	0.659	0.880	0.641	0.719
WaRPAD	**0.986**	**0.998**	0.991	0.810	0.940	0.936	**0.981**	0.924	0.946
RDD **(ours)**	0.926	0.997	**0.996**	0.983	**1.000**	**0.999**	0.946	**0.999**	**0.981**

Table 2. **AI-generated image detection performance (AUROC) of RDD and baselines in the LSUN-Bedroom [18] benchmark. Bold** and underline denote the best method and the second best methods, respectively.

Method	ADM	DDPM	Diff-PjGAN	Diff-StyleGAN2	IDDPM	LDM	PNDM	PrGAN	PjGAN	StGAN	Mean
Training-based Methods											
FatFormer	0.745	0.709	0.998	1.000	0.824	0.944	0.999	1.000	0.999	0.988	0.921
AIDE	0.636	0.722	0.860	0.951	0.679	0.807	0.941	0.899	0.910	0.840	0.825
Training-free Methods											
RIGID	0.742	0.887	0.937	0.914	0.855	0.846	0.843	0.957	0.944	0.681	0.861
MINDER	0.706	0.796	0.973	0.942	0.782	0.844	0.896	0.970	0.973	0.805	0.869
AEROBLADE	0.556	0.728	0.417	0.403	0.656	0.601	0.311	0.387	0.418	0.288	0.476
Manifold Bias	0.784	0.900	0.967	0.944	0.885	0.927	0.901	**0.996**	0.977	0.915	0.920
WaRPAD	0.785	0.937	**0.988**	0.965	0.908	**0.940**	**0.970**	0.995	**0.986**	0.870	0.934
RDD **(ours)**	**0.823**	**0.978**	0.935	**0.967**	**0.964**	0.921	0.938	0.983	0.956	**0.937**	**0.940**

Second, LSUN-Bedroom [19] mainly evaluates on GAN and image-diffusion-based generative models against real LSUN [29] data, where the image is all resized to 256 $\times$256. We refer to the Appendix for the details.

Baselines We consistently compare against reproducible training-free baselines: AEROBLADE [19], Manifold induced Bias [2], RIGID [6], MINDER [23], and WaRPAD [3]. We reproduce AEROBLADE, Manifold-induced Bias, and WarPAD from the authors' official code. RIGID and MINDER utilize a pre-trained DINOv2 [14] model, reproducing the results with the provided hyperparameters. Furthermore, for reference, we also test the cross-domain performance of training-based detectors: AIDE [27] and FatFormer [10]. Specifically, the detectors are evaluated by testing the pre-trained checkpoint in a zero-shot manner.

Implementation Details The performance of all methods is reported by the area under the ROC curve (AUROC). We use three types of diffusion models for ensemble: SDv1.4/v2-base, and MiniSD [9], and apply the same for AEROBLADE. We fix all hyperparameters of RDD across different benchmarks. Specifically, we set $t = 0.05$ and $\lambda_{\mathcal{R}} = 0.5$. We implement our code in the Pytorch [15] framework via a single A100 GPU. We follow [19] for the augmentation of the input image. Different from the AEROBLADE, we average the score while computing the S_{image} score.

4.2. Experiment Results

We present the main experiment result in Table 1 and 2. RDD consistently achieves the best performance on all benchmarks on average. Moreover, our method achieves the best or second-best performance in 16 out of 18 benchmarks, supporting the efficacy and generalizability of our unified framework.

In contrast, methods relying only on AE or the latent-level diffusion model struggle with specific benchmarks.

Table 3. **AIGI detection performance of each component of our RDD.** We also include the performance of f_{AE} and f_{latent}.

Benchmark	GenImage	LSUN-Bedroom
f_{latent}	0.408	0.666
f_{AE}	0.902	0.476
S_{latent}	0.667	0.963
S_{image}	0.969	0.464
S_{RDD}	0.981	0.940

Table 4. **AIGI detection performance of RDD with respect to different integrations.** We measure AUROC in the GenImage and LSUN-Bedroom benchmark. λ_A and λ_M corresponds to integration via addition and multiplication, respectively.

Benchmark	$\lambda_A = 1$	$\lambda_A = 10^{-2}$	$\lambda_A = 10^{-4}$	$\lambda_M = 1$	$\lambda_M = 2$	$\lambda_M = 3$
GenImage	0.673	0.793	0.973	0.986	0.981	0.971
LSUN-Bedroom	0.965	0.965	0.515	0.858	0.940	0.957

Table 5. **AIGI detection performance of RDD with respect to the size and σ of the Gaussian kernel $\mathcal{F}$ in the GenImage benchmark.** Note that kernel size and std of 3 and 0.8 correspond to our default choice, respectively.

σ	3	5	7
0.5	0.970	0.966	0.966
0.8	0.981	0.980	0.981
1.1	0.982	0.980	0.980
1.4	0.982	0.979	0.978

For example, Manifold Bias struggles in detecting LDM-generated images. On the other hand, AEROBLADE struggles where the underlying AE model is not given (*e.g.*, ADM), or GAN-based methods in LSUN-Bedroom benchmarks. We further show in Figure 7 for the edge cases of each method. While RIGID and MINDER show moderate performance overall, our unified RDD consistently outperforms them on average.

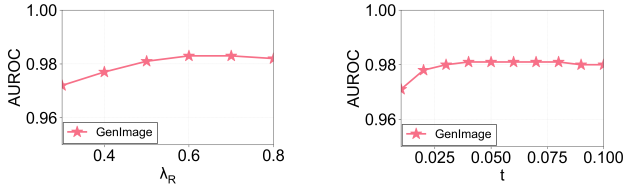
Finally, training-based methods show great performance on the benchmarks that are similar to their training data distribution; ImageNet and SD-generated data for AIDE and LSUN, and ProGAN-generated data for Fatformer; they underperform over several training-free methods on unobserved data distribution.

4.3. Analysis

Contribution of Each Component Since our RDD consists of two independent distance functions, we analyze the individual performance of each S_{latent} and S_{image} independently. We show the results in Table 3. First, each of our proposed debiased scores S_{latent} and S_{image} improves over f_{Latent} and f_{AE}, respectively. Moreover, we observe that two

(a) (b)

Figure 7. **Edge cases. (a)** Real ImageNet images where f_{AE} assign lowest reconstruction error. **(b)** Real ImageNet images where f_{latent} assigns highest reconstruction error.



(a) λ_R (b) t

Figure 8. **Hyperparameter analysis of RDD.** AUROC performance on the GenImage benchmark to the hyperparameter λ_R (**(a)**) and t (**(b)**).

scores are complementary: S_{latent} is efficient where S_{image} struggles, and vice versa. Finally, we observe that S_{RDD} is competitive against the best of S_{latent} and S_{image}, even showing synergistic gain in the GenImage benchmark.

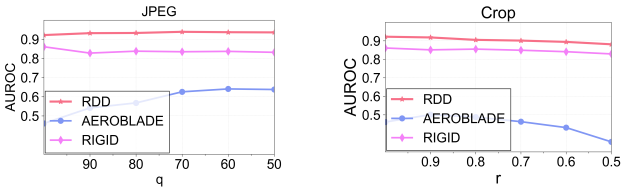
Furthermore, we also explore alternatives on the combination of S_{image} and S_{latent}. For the other choice, we explore addition: $S_{\text{image}} + \lambda_A S_{\text{latent}}$ or multiplication: $S_{\text{image}} \times S_{\text{latent}}^{\lambda_M}$. We show the result in Table 4. Addition-based integration fails to balance between the latent-based debiasing and image-based debiasing methods.

Hyperparameter Analysis We explore the effect of two major hyperparameters, the diffusion timestep t and normalizing hyperparameter λ_R. We explore their individual effects in the GenImage benchmark. We show the result in Figure 8. While RDD's performance underperforms in small λ_R or t, the overall performance is robust to the change of hyperparameters otherwise. Furthermore, we also analyze the performance of RDD with respect to the size and standard deviation of the Gaussian kernel applied for low-pass filtering. We show the result in Table 5, where RDD is stable across various hyperparameters. Furthermore, we test alternative rotation angles for comparison: 45-degree and 180-degree rotation, which show 0.983/0.984 on the GenImage benchmark, respectively.

Robustness to Perturbations Web-scale images undergo various perturbations, including JPEG compression and resolution changes. Hence, we further test our RDD in the

Table 6. **Belonging vs non-belonging image detection performance (AUROC) of S_{image} compared to LatentTracer [26].** We denote $\mathcal{M}_1$ and $\mathcal{M}_2$ to the belonging and non-belonging model, respectively. **Bold** denotes the best method.

	$\mathcal{M}_1$: SDv1.5			$\mathcal{M}_1$: SDv2-base			$\mathcal{M}_1$: Kandinsky		
Method	SDv2-base	SDv2.1	Kand	SDv1.5	SDv2.1	Kand	SDv1.5	SDv2-base	SDv2.1
LatentTracer	0.9990	0.9981	0.9974	0.9993	0.9988	0.9982	0.9962	0.9960	0.9930
AEROBLADE	0.9677	0.9932	0.8270	0.9581	0.9939	0.8467	0.9946	**0.9973**	0.9978
S_{image} **(ours)**	**1.0000**	**1.0000**	**1.0000**	**0.9996**	**1.0000**	**0.9991**	**0.9976**	0.9972	**0.9979**



(a) JPEG (b) Crop

Figure 9. **Robustness of RDD under perturbation.** AUROC performance on the LSUN-Bedroom benchmark to the JPEG compression **(a)** and center-cropping and resizing **(b)**. We also test AEROBLADE and RIGID at the same level of corruption.

perturbation of input images. For the perturbation, we select JPEG compression and the center cropping operation. We show our results in the LSUN-Bedroom benchmark. For comparison, we include the robustness of the baselines, AEROBLADE and RIGID, under such corruption. We show the results in Figure 9. RDD remains competitive despite the image corruption.

Alternative Augmentations While our low-pass filtering and rotation-based augmentation schemes prove their efficacy in the benchmarks, other augmentations can be applied for the construction of adversarial images. For comparison, we also test `RandomResizedCrop` and `ColorJitter`. While the hyperparameter search space of the two methods is massive, we instead test the standard choice widely used in self-supervised image augmentation [14]. We will further discuss the details in the Appendix. For `ColorJitter` and `RandomResizedCrop`, results are 0.907/0.841 in the GenImage benchmark, which underperform over our augmentation.

Extension to LDM-generated image attribution. Compared to f_{AE}, our S_{image} score can effectively detect LDM-generated images in near-perfect performance when the AE of the inspected model is given (*e.g.*, SD1.4/v2-base, Midjourney). This motivates us to directly apply our S_{image} score in the attribution of LDM-generated images. To be specific, a recent line of works [25, 26] design an uncertainty score function $S(\mathbf{x})$ that distinguishes the belonging data $\mathbf{x}$ generated from $\mathcal{M}_1$ (*i.e.*, $S(\mathbf{x}) \leq \tau$) and the non-belonging data generated from the other generative model $\mathcal{M}_2$ (*i.e.*, $S(\mathbf{x}) > \tau$). We denote such task as "$\mathcal{M}_1$ `vs` $\mathcal{M}_2$" model attribution task.

We also apply S_{image} on detecting model-generated images. For a given LDM model $\mathcal{M}_1$, we directly apply $S_{\text{image}}(\mathbf{x})$ to distinguish the belonging images generated from $\mathcal{M}_1$ from the others.

We follow the practice of Wang et al. [26] where 540 different images are generated by 54 prompts with 10 images per prompt for a given LDM. 4 different LDMs, SDv1.5, SDv2-base, SDv2.1, and Kandinsky [17], are applied to generate the dataset. We also compare against LatentTracer [26], the baseline that performs input optimization at test time. We test LatentTracer on the official code released by the authors. We report the performance of S_{image}, and AEROBLADE. For S_{image}, we set $\lambda_R = 0.25$.

We report the result in Table 6. Our proposed S_{image} improves over LatentTracer in 9 tasks. On the other hand, AEROBLADE struggles in "`SD vs Kandinsky`" tasks. It is worth noting that S_{image} is much more efficient in time complexity than LatentTracer. Namely, S_{image} takes 0.292s per sample and achieves 50x speedup against the LatentTracer, which takes 14.65s/sample in a single A100 gpu,

5. Limitations and Discussion

Our method requires both autoencoder-based reconstruction in augmented image data and network evaluation in augmented latent, which can be slower than baseline methods.

6. Conclusion

This paper discusses the extension of LDM to training-free AI-generated image detection by leveraging the reconstruction error in both the image and latent spaces. Furthermore, we propose a debiasing strategy based on the augmentation operation, which cancels out the reconstruction loss measured in the off-manifold data. The biggest strength of our framework is that the constructed methods can be unified naturally with minimal introduction of hyperparameters. Finally, our debiased metric can be applied to the attribution of LDM-generated images.

Acknowledgment

This work was in part supported by Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.RS-2025-25442149, LG AI STAR Talent Development Program for Leading Large-Scale Generative AI Models in the Physical AI Domain, 50%). This work is also supported by LG AI Research.

References

[1] Blake Brittain. Google sued by us artists over ai image generator, 2024. 1

[2] Jonathan Brokman, Amit Giloni, Omer Hofman, Roman Vainshtein, Hisashi Kojima, and Guy Gilboa. Manifold induced biases for zero-shot and few-shot detection of generated images. In *ICLR*, 2025. 2, 6, 10

[3] Sungik Choi, Hankook Lee, and Moontae Lee. Training-free detection of ai-generated images via cropping robustness. In *NeurIPS*, 2025. 6

[4] Davide Cozzolino, Giovanni Poggi, Matthias Nießner, and Luisa Verdoliva. Zero-shot detection of ai-generated images. In *ECCV*, 2024. 2

[5] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 3, 4, 5

[6] Zhiyuan He, Pin-Yu Chen, and Tsung-Yi Ho. Rigid: A training-free and model-agnostic framework for robust ai-generated image detection. *arXiv preprint arXiv:2405.20112*, 2024. 2, 6

[7] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. In *ICLR*, 2018. 4

[8] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. In *ICLR*, 2014. 3

[9] Lambda Labs. miniSD-diffusers: A Text-to-Image Model based on Stable Diffusion, 2022. 6

[10] Huan Liu, Zichang Tan, Chuangchuang Tan, Yunchao Wei, Yao Zhao, and Jingdong Wang. Forgery-aware adaptive transformer for generalizable synthetic image detection. In *CVPR*, 2024. 6

[11] Timo Lüddecke and Alexander Ecker. Image segmentation using text and image prompts. In *CVPR*, 2022. 3

[12] Yunpeng Luo, Junlong Du, Ke Yan, and Shouhong Ding. Lare^2: Latent reconstruction error based method for diffusion-generated image detection. In *CVPR*, 2024. 2

[13] MidJourney. MidJourney AI Art Generator, 2022. 4

[14] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel HAZIZA, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024. 6, 8

[15] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *NeurIPS*, 2019. 6

[16] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 2

[17] Anton Razzhigaev, Arseniy Shakhmatov, Anastasia Maltseva, Vladimir Arkhipkin, Igor Pavlov, Ilya Ryabov, Angelina Kuts, Alexander Panchenko, Andrey Kuznetsov, and Denis Dimitrov. Kandinsky: An improved text-to-image synthesis with image prior and latent diffusion. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 2023. 8

[18] Jonas Ricker, Simon Damm, Thorsten Holz, and Asja Fischer. Towards the detection of diffusion model deepfakes. In *VISAPP*, 2024. 6, 10

[19] Jonas Ricker, Denis Lukovnikov, and Asja Fischer. Aeroblade: Training-free detection of latent diffusion images using autoencoder reconstruction error. In *CVPR*, 2024. 1, 2, 3, 6

[20] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 1, 3, 4

[21] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. Laion-5b: An open large-scale dataset for training next generation image-text models. In *Advances in Neural Information Processing Systems, Datasets and Benchmarks Track*, 2022. 3

[22] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015. 5

[23] Chung-Ting Tsai, Ching-Yun Ko, I-Hsin Chung, Yu-Chiang Frank Wang, and Pin-Yu Chen. Understanding and improving training-free ai-generated image detections with vision foundation models. *arXiv preprint arXiv:2411.19117*, 2024. 2, 6

[24] Zhendong Wang, Jianmin Bao, Wengang Zhou, Weilun Wang, Hezhen Hu, Hong Chen, and Houqiang Li. Dire for diffusion-generated image detection. In *ICCV*, 2023. 2

[25] Zhenting Wang, Chen Chen, Yi Zeng, Lingjuan Lyu, and Shiqing Ma. Where did i come from? origin attribution of ai-generated images. In *NeurIPS*, 2023. 8

[26] Zhenting Wang, Vikash Sehwag, Chen Chen, Lingjuan Lyu, Dimitris N. Metaxas, and Shiqing Ma. How to trace latent generative model generated images without artificial watermark? In *ICML*, 2024. 8, 10

[27] Shilin Yan, Ouxiang Li, Jiayin Cai, Yanbin Hao, Xiaolong Jiang, Yao Hu, and Weidi Xie. A sanity check for ai-generated image detection. In *ICLR*, 2025. 6

[28] Hyunsu Yim. South korea to criminalise watching or possessing sexually explicit deepfakes, 2024. 1

[29] 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. *https://arxiv.org/abs/1506.03365*, 2015. 6

[30] Runjia Zeng, Cheng Han, Qifan Wang, Chunshu Wu, Tong Geng, Lifu Huangg, Ying Nian Wu, and Dongfang Liu. Visual fourier prompt tuning. In *NeurIPS*, 2024. 4

[31] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018. 3

[32] Mingjian Zhu, Hanting Chen, Qiangyu Yan, Xudong Huang, Guanyu Lin, Wei Li, Zhijun Tu, Hailin Hu, Jie Hu, and Yunhe Wang. Genimage: A million-scale benchmark for detecting ai-generated image. In *NeurIPS*, 2023. 4, 5, 6, 10

A. Further information on the main experiment settings.

GenImage. GenImage benchmark consists of real images and latent-diffusion-model-based AI-generated images, where each set consists of the same number of real and AI-generated images. We download GenImage data from the author's official repository [32].

Deepfake-LSUN-Bedroom. The Deepfake-LSUN-Bedroom benchmark consists of 10000 real LSUN-Bedroom images and 100000 AI-generated images that consist mostly of GANs, but also include several diffusion models (LDM, PNDM, DDPM). We also downloaded the Deepfake-LSUN-Bedroom benchmark from the author's official repository [18].

Implementations. We follow the author's implementation for the AEROBLADE [1] and Manifold Bias [2], respectively. We reproduce the RIGID and MINDER following the official hyperparameters of the author's setting and augmentation based on the DINOv2 test-time evaluation settings, which involves resizing and center cropping the given input image. RDD, RIGID, and Manifold Bias involve the sampling of the random noise function. Given that, we average the results in 3 random seeds for those methods. All experiments are done in a single A100 gpu under the Pytorch 2.2.1+cu221 version in the Linux OS. All the memory is sufficient to operate in the 40GB A100 gpu, with the biggest memory applied in Manifold Bias of 38GB, which applies the LLAVA-HF model to caption the given input image. In contrast, we neglect the effect of the captioning model and instead use a null text-prompt for efficiency, which still outperforms the Manifold Bias method. Finally, when calculating the S_{latent}, we use the decoder output distance following Brokman et al. [2].

B. Ablation on SSL augmentations

For the use of ColorJitter and RandomResizedCrop, we use the default augmentation hyperparameter in the DINOv2 setting.

C. Further information in the LDM-generated image attribution

As mentioned in the main section, we generate the data given the 54 prompts with 10 iterations each, as noted in the official github repository of LatentTracer [26].

[1] https://github.com/jonasricker/aeroblade
[2] https://tinyurl.com/zeroshotimplementation