$\boldsymbol{Z}_T$: SIMPLIFYING DATA DISTRIBUTIONS FOR EFFECTIVE AI-GENERATED IMAGE DETECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

The rapid progress of image generation technology has heightened concerns about misuse, including misinformation and deepfakes, creating an urgent need for reliable AI-generated image (AIGI) detection. Although state-of-the-art detectors based on inversion and reconstruction achieve strong performance in the ideal setting, where the inversion and reconstruction models are the same, their performance drops significantly when these two models differ. All previous works on diffusion-based AIGI detection overlook an important prior, that the noise of generated images from various models is drawn from a standard Gaussian distribution $\mathcal{N}(\mathbf{0}, \boldsymbol{I})$. We propose to exploit this prior in order to improve the generalization and robustness of AIGI detection. We first perform a series of analyses on the distributional dynamics of the diffusion inversion process. Our analysis shows that the prediction errors of the denoising model accumulate at each inversion step, preventing the real image distribution converging to $\mathcal{N}(\mathbf{0}, \boldsymbol{I})$ in the noise space. To further deepen our analysis, we employ a Chi-squared test and a Bayes classifier to identify the distributions to which real and generated image patches converge. Our results show that the distributions of generated and real image patches converge to two very simple forms: the former approximates $\mathcal{N}(\mathbf{0}, \boldsymbol{I})$, while the latter becomes a mixture of a non-standard Gaussian distribution and a distribution with small $\|\boldsymbol{z}_T\|$. Motivated by these findings, we propose Inversion Noise Classification (INC), a straightforward classification method that operates directly on $\boldsymbol{z}_T$. Extensive experiments across diverse diffusion models and settings show that INC consistently outperforms state-of-the-art AIGI detectors, with notable superiority in robustness to input distortions and generalizability across inversion methods.

1 INTRODUCTION

AI generated images (AIGI) are already omnipresent, consumed daily by billions. It is in your social media feeds (Moller et al., 2026), your two-factor authentication (Zukanovic), every photo you may ever take (Germain). Even darker, it is used to lure thousands into forced labor scams (Burke-Preputnik et al.) and human trafficking networks (United States Department of State & in Persons). Compounding this, human detection accuracy for AIGI hovers near chance (Pavão, 2025; Roca et al., 2025), necessitating robust and generalizable detection methods. In this paper, we seek a mechanistic explanation of the process by which AIGI is made detectable, grounded in the generative process itself.

Latent diffusion models (LDMs) have emerged as the dominant paradigm for high-fidelity image synthesis, underpinning widely deployed systems such as Stable Diffusion (SD) and its derivatives. Their capacity to generate photorealistic content at scale has necessitated the development of robust and generalizable detection methods. Furthermore, detecting AIGI is difficult precisely because modern generative models are optimized to be indistinguishable from reality. Robust and generalizable detection requires moving beyond surface-level features to understand the underlying mechanics of image generation.

A natural approach to principled AIGI detection is to interrogate images using the generative process itself. The core intuition, shared by several recent methods, is that diffusion models reconstruct their own outputs more faithfully than they reconstruct real images. DIRE (Wang et al., 2023) exploits this by measuring reconstruction error after a full cycle of inversion and resampling. SeDID (Ma et al.,

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.

2023) refines this idea by examining step-wise reconstruction error along the inversion trajectory. Aeroblade (Ricker et al., 2024) takes a complementary approach, using the reconstruction error of the VAE encoder-decoder alone, without invoking the diffusion process. Despite their empirical success, none of these methods provide a mechanistic understanding of why real and generated images should traverse distinct trajectories, a gap this paper takes steps toward addressing.

We show theoretically that small errors in noise prediction compound and accumulate with each step of diffusion inversion, and that this error propagation differs systematically between real and generated images. Additionally, we empirically verify that the distributions obtained by inversion to $t = T$ are relatively simple and more amenable to classification across diverse model families (Friedman, 1997; Attoor & Dougherty, 2004; Refinetti et al., 2023), particularly when compared to the complexity of the data manifold at $t = 0$ (Biroli et al., 2024; Ventura et al., 2025). Together, these observations clarify a mechanism by which real and AIGI distributions separate and simplify throughout the inversion process: deterministic inversion drives generated images towards a noise distribution approximating $\mathcal{N}(\mathbf{0}, \boldsymbol{I})$ while real images deviate from this target prior, rendering real and generated images as distinct, separable populations in noise space.

Motivated by this analysis, we propose Inversion Noise Classification (INC), performing AIGI detection directly on the fully inverted noise representation, $\boldsymbol{z}_T$, rather than in pixel or latent reconstruction space. Classification in $\boldsymbol{z}_T$ is more robust than reconstruction-based approaches across a range of real-world image distortions, including JPEG compression, blurring, and resizing, conditions under which prior methods degrade significantly. We support this claim with a broad suite of experiments across multiple latent diffusion models.

Our contributions are as follows:

- Inversion error theory: We derive a theoretical account of how noise prediction errors compound and accumulate during diffusion inversion. We show that the discrepancy between real and generated inversion trajectories is governed by the accumulated history of real-image-specific mismatch terms, explaining why late inversion steps are especially informative for AIGI detection.
- Distributional analyses: We characterize the statistical properties of $\boldsymbol{z}_T$ for real and generated images through analyses including χ^2-based patch-energy statistics and predicted-noise mean and covariance statistics. We further show that even a simple Bayes-based classifier, *Quadratic Discriminant Analysis* (QDA), achieves strong detection performance on $\boldsymbol{z}_T$. These results show that DDIM inversion transforms the original latent space into a simpler and more prior-aligned space at $t = T$, while preserving a clear distributional gap between real and generated samples. This supports our claim that $\boldsymbol{z}_T$ is distributionally simpler and provides a tractable basis for AIGI detection across diverse diffusion models.
- Inversion Noise Classification (INC): We propose classifying images based on their fully-inverted noise representations, $\boldsymbol{z}_T$, and demonstrate that this approach achieves superior performance across diffusion model types, stronger robustness against distortions, and better generalization to unseen data distributions compared to other diffusion-based AIGI detection methods.

2 Related Works

Modern diffusion models have their roots in Langevin dynamics SDEs (Jarzynski, 1997a;b) and their discrete analogue in annealed sampling (Neal, 1998); initial research on diffusion models (Sohl-Dickstein et al., 2015; Ho et al., 2020; Song & Ermon, 2019) cite these directly and are primarily concerned with finding feasible score-function objectives and methods for sampling. Later works explore alternative formulations for noise schedules (Song et al., 2020; Karras et al., 2022; Lin et al., 2024), training objective (Liu et al., 2023; Salimans & Ho, 2022; Kingma et al., 2023; Song et al., 2023; Liu et al., 2024), architecture (Peebles & Xie, 2023), and sampling method (Song et al., 2021; Kingma & Gao, 2023; Lu et al., 2022; 2025; Wang et al., 2025; Xu et al., 2025). Additional developments in diffusion introduce popular features, including conditionality (Dhariwal & Nichol, 2021; Ho & Salimans, 2021; Brooks et al., 2023; Labs et al., 2025) and latent-space processing (Rombach et al., 2021; Luo et al., 2024a; Esser et al., 2024). Our work builds specifically on latent diffusion models with deterministic samplers.

Deterministic inversion is the process of mapping images to an approximate noise-space representation by running the sampling process in reverse (Song et al., 2021; Dhariwal & Nichol, 2021). It underpins a broad class of editing and personalization methods that encode source images to noise before redirecting the resampling process toward the desired edit (Mokady et al., 2023; Ju et al., 2024; Miyake et al., 2025). Beyond editing, inversion is a useful analytical tool: the noise representations it produces expose distributional structure that is not apparent in image space. Inverted representations of images are known to deviate from the standard Gaussian target (Hong et al., 2024; Parmar et al., 2023; Garibi et al., 2024) and to carry structured statistical information reflecting their source images (Staniszewski et al., 2025). These studies suggest that noise space encodes meaningful differences between image populations. We exploit this directly, proposing classification in the inverted noise representation as a principled method for AIGI detection.

AIGI detection has been approached from many angles, including statistical properties of AIGI distributions (Luzi et al., 2024; Jia et al., 2025; Zisman & Shaham, 2026), pixel-domain artifact analysis (Joslin et al., 2024; Sha et al., 2023), frequency-domain signatures (Frank et al., 2020; Alam et al., 2024; Lu & Ebrahimi, 2024; Ricker et al., 2022), feature-space classification (Ojha et al., 2023; Zhong et al., 2026), and detection by multimodal large language models (Huang et al., 2025; Tan et al., 2025; Zhou et al., 2025). More recently, attention has turned to applying the generative process of diffusion models themselves to AIGI detection. Diffusion denoising trajectories enjoy well-studied theoretical properties: prior works have linked learned trajectories to optimal transport maps (Khrulkov et al., 2023; Liu, 2022; Ghimire et al., 2023) and theorized that noise prediction approximation error accumulates with each sampling step during generation (Li & van der Schaar, 2024). Together, these studies suggest that images well-matched to the model's learned distribution (*i.e.*, its own outputs) are less susceptible to prediction error and are therefore handled more faithfully by the diffusion model. This intuition is exploited by recent reconstruction-based AIGI detection methods (Ma et al., 2023; Wang et al., 2023; Ricker et al., 2024; Ameen & Islam, 2026), which differ primarily in where reconstruction error is measured. Other works extend reconstruction-based AIGI detection with contrastive learning (Chen et al., 2024), error refinement (Luo et al., 2024b), frequency-space detection (Chu et al., 2025), and feature space analysis (Liang et al., 2025). These methods are primarily empirical, offering limited insight into why real and generated images should traverse distinct trajectories. Furthermore, none exploit the distributional properties of the fully-inverted representation $\boldsymbol{z}_T$. This paper takes steps toward addressing each of these gaps.

Classification becomes more difficult as data distributions grow more complex, while both classical and neural classifiers tend to benefit from simpler, lower-order statistical structure (Friedman, 1997; Mangalam & Prabhu, 2019; Zhi-Qin et al., 2020; Refinetti et al., 2023; Belrose et al., 2024). This suggests that simpler distributions are more amenable to accurate classification. Crucially, the $\mathcal{N}(\mathbf{0}, \boldsymbol{I})$ noise prior shared by most diffusion models provides a simple and generalizable space for AIGI detection across diverse model families.

3 PRELIMINARIES

We briefly summarize the latent diffusion and DDIM inversion formulation used in this work. Latent diffusion models operate in a VAE latent space rather than directly in pixel space (Rombach et al., 2021). Their dynamics can be expressed in several equivalent parameterizations, but in our theory work in this paper, we work exclusively with the noise-prediction form and the DDIM inversion process (Ho et al., 2020; Song & Ermon, 2019; Song et al., 2020; 2021), since this is the standard form used by DDIM and it allows the inversion mismatch to be modeled and analyzed directly.

Let $\mathcal{Z} \subset \mathbb{R}^D$ denote the latent space, and consider discrete timesteps $t \in \{0, 1, \ldots, T\}$. For a data sample $\boldsymbol{x} \sim p_{\text{data}}(\boldsymbol{x})$, let $\boldsymbol{z}_0 = Enc(\boldsymbol{x})$ denote its VAE latent representation. We adopt the standard latent noising convention $\boldsymbol{z}_t = \alpha_t \boldsymbol{z}_0 + \sigma_t \boldsymbol{\varepsilon}$, $\boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$, where $\alpha_t \in [0, 1]$ and $\sigma_t \in [0, 1]$ are the signal and noise schedule coefficients. Typically, $\alpha_0 \approx 1$ and $\sigma_0 \approx 0$, while $\alpha_T \approx 0$ and $\sigma_T \approx 1$.

We denote the network prediction by $\boldsymbol{\varepsilon}_\theta(\boldsymbol{z}_t, t)$. The deterministic DDIM update derived by Song et al. (2021) is

$$\frac{\boldsymbol{z}_s}{\alpha_s} = \frac{\boldsymbol{z}_t}{\alpha_t} + \boldsymbol{\varepsilon}_\theta(\boldsymbol{z}_t, t) \left(\frac{\sigma_s}{\alpha_s} - \frac{\sigma_t}{\alpha_t} \right), \tag{1}$$

where sampling uses reverse time ($s \leq t$), while DDIM inversion uses the same rule in forward time ($s \geq t$). As noted in Song et al. (2021), DDIM can be interpreted as a first-order Euler discretization

of the corresponding deterministic ODE, and higher-order solvers also exist (Lu et al., 2022; 2025; Liu et al., 2023; Wang et al., 2025).

Since our later analysis is discrete, we rewrite the DDIM inversion step from timestep t to $t+1$ in the increment form

$$\Delta \boldsymbol{z}_t := \boldsymbol{z}_{t+1} - \boldsymbol{z}_t = a_t \boldsymbol{z}_t + b_t\, \boldsymbol{\varepsilon}_\theta(\boldsymbol{z}_t, t), \tag{2}$$

where

$$a_t := \frac{\alpha_{t+1}}{\alpha_t} - 1, \qquad b_t := \sigma_{t+1} - \frac{\alpha_{t+1}}{\alpha_t}\sigma_t. \tag{3}$$

This discrete noise-prediction DDIM form will be used throughout the analysis section.

4 Analysis

4.1 The Accumulative Error

We now analyze the real-vs-generated discrepancy under the discrete DDIM inversion formulation introduced in Section 3. For notational convenience, let $\boldsymbol{\varepsilon}_t^\star$ denote a reference noise prediction at timestep t. Specifically, $\boldsymbol{\varepsilon}_t^\star$ represents the predicted noise that would be used along an idealized reference inversion trajectory under the same scheduler. We then model the actual predicted noises for generated and real inputs as deviations from this common reference. We model the actual predicted noises for generated and real inputs as

$$\widehat{\boldsymbol{\varepsilon}}_t^{\text{gen}} = \boldsymbol{\varepsilon}_t^\star + \boldsymbol{\delta}_t, \qquad \widehat{\boldsymbol{\varepsilon}}_t^{\text{real}} = \boldsymbol{\varepsilon}_t^\star + \boldsymbol{\delta}_t + \boldsymbol{\gamma}_t, \tag{4}$$

where $\boldsymbol{\delta}_t$ is the DDIM noise approximation error and $\boldsymbol{\gamma}_t$ is the additional mismatch term for real images. Accordingly, the generated and real DDIM updates are

$$\boldsymbol{z}_{t+1}^{\text{gen}} = (1+a_t)\boldsymbol{z}_t^{\text{gen}} + b_t\, \widehat{\boldsymbol{\varepsilon}}_t^{\text{gen}}, \qquad \boldsymbol{z}_{t+1}^{\text{real}} = (1+a_t)\boldsymbol{z}_t^{\text{real}} + b_t\, \widehat{\boldsymbol{\varepsilon}}_t^{\text{real}}. \tag{5}$$

If there is no noise approximation error, *i.e.,* $\boldsymbol{\delta}_t = \mathbf{0}$, the terminal latents $\boldsymbol{z}_T^{\text{gen}} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$. We define the latent gap

$$\boldsymbol{e}_t := \boldsymbol{z}_t^{\text{real}} - \boldsymbol{z}_t^{\text{gen}}. \tag{6}$$

Here, $\boldsymbol{e}_0$ is the initial latent gap, while $\boldsymbol{\gamma}_0$ is the additional mismatch at timestep 0, which enters the first DDIM update from 0 to 1.

Proposition 4.1 (The Accumulative Error of DDIM Inversion)**.** *Under the standard monotone schedule, we have $\alpha_{t+1} < \alpha_t$, $\sigma_{t+1} > \sigma_t$. Therefore, $1 + a_t = \frac{\alpha_{t+1}}{\alpha_t} \in (0,1)$, $a_t < 0$, $b_t = \sigma_{t+1} - \frac{\alpha_{t+1}}{\alpha_t}\sigma_t > 0$. Hence, in the latent-gap recursion below, the previous gap is propagated with a contraction factor $1 + a_t \in (0,1)$, while each new mismatch term enters with a positive coefficient b_t. Then the latent gap defined in Eq. 6 satisfies the recursion*

$$\boldsymbol{e}_{t+1} = (1+a_t)\boldsymbol{e}_t + b_t\boldsymbol{\gamma}_t. \tag{7}$$

Consequently, the terminal gap at timestep T can be written as

$$\boldsymbol{e}_T = \left(\prod_{j=0}^{T-1}(1+a_j)\right)\boldsymbol{e}_0 + \sum_{s=0}^{T-1}\left(\prod_{j=s+1}^{T-1}(1+a_j)\right) b_s\boldsymbol{\gamma}_s. \tag{8}$$

Hence, $\boldsymbol{e}_T$ contains the accumulated contribution of all extra real-image mismatch terms $\boldsymbol{\gamma}_0, \boldsymbol{\gamma}_1, \ldots, \boldsymbol{\gamma}_{T-1}$ up to timestep T.

Proposition 4.1 gives recursion-based characterization of the latent gap, showing that $\boldsymbol{e}_t$ depends on the full history of extra $\boldsymbol{\gamma}_t$ up to the chosen inversion timestep. It does not imply that $\|\boldsymbol{e}_t\|$ must increase monotonically with t, since the contraction factor $(1+a_t) \in (0,1)$ and the direction of $\boldsymbol{\gamma}_t$ may cause partial cancellation. Importantly, in Eq. 8, the coefficient of the initial gap $\boldsymbol{e}_0$ becomes very small for large T. Therefore, the effect of the initial image content is suppressed at late timesteps, while the accumulated mismatch term becomes more important. Motivated by this, we perform statistical analyses the mean and covariance of the predicted-noise distributions to examine how the $\boldsymbol{z}_T^{real}$ distribution deviates from the $\boldsymbol{z}_T^{gen}$ and from the Gaussian prior.

4.2 Empirical Mean and Covariance Analysis of the Predicted-noise Distributions

The error accumulation effect is reflected in the empirical statistics of the predicted noise. Let $\boldsymbol{\mu}_t^{\varepsilon,\text{real}} := \mathbb{E}[\widehat{\boldsymbol{\varepsilon}}_t^{\text{real}}]$, $\boldsymbol{\mu}_t^{\varepsilon,\text{gen}} := \mathbb{E}[\widehat{\boldsymbol{\varepsilon}}_t^{\text{gen}}]$, and $\Sigma_t^{\varepsilon,\text{real}} := \text{Cov}(\widehat{\boldsymbol{\varepsilon}}_t^{\text{real}})$, $\Sigma_t^{\varepsilon,\text{gen}} := \text{Cov}(\widehat{\boldsymbol{\varepsilon}}_t^{\text{gen}})$. Fig. 1 (a) and (b) show that, as the inversion timestep increases, both the predicted-noise mean gap $\|\Delta\boldsymbol{\mu}_t^{\epsilon}\|_2 = \|\boldsymbol{\mu}_t^{\varepsilon,\text{real}} - \boldsymbol{\mu}_t^{\varepsilon,\text{gen}}\|_2$ and the predicted-noise covariance gap $\|\Delta\Sigma_t^{\epsilon}\|_F = \|\Sigma_t^{\varepsilon,\text{real}} - \Sigma_t^{\varepsilon,\text{gen}}\|_F$ increase substantially before gradually saturating at late timesteps. This indicates that, as inversion proceeds, the predicted-noise distributions of real and generated images become increasingly different in their first- and second-order statistics.

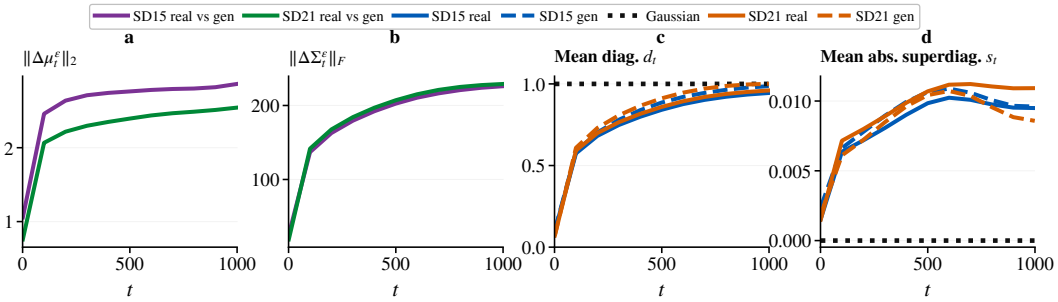


Figure 1: Predicted-noise distribution statistics across DDIM inversion. **(a)** $\|\Delta\boldsymbol{\mu}_t^{\epsilon}\|_2$ and **(b)** $\|\Delta\Sigma_t^{\epsilon}\|_F$ increase before saturating, showing that the separation between $\widehat{\boldsymbol{\varepsilon}}_t^{\text{real}}$ and $\widehat{\boldsymbol{\varepsilon}}_t^{\text{gen}}$ becomes more pronounced as inversion proceeds. **(c)** Main-diagonal and **(d)** first-superdiagonal covariance statistics.

Fig. 1 (c) and (d) further support this trend by examining both the main diagonal and the first superdiagonal of the predicted-noise covariance. Define $d_t := \frac{1}{D}\sum_{i=1}^{D}(\Sigma_t^{\varepsilon})_{ii}$ and $o_t := \frac{1}{D-1}\sum_{i=1}^{D-1}\left|(\Sigma_t^{\varepsilon})_{i,i+1}\right|$. Fig. 1 (c) shows that both d_t^{real} and d_t^{gen} increase toward the Gaussian-reference value 1 as t grows and the difference becomes more pronounced at later time steps. Fig. 1 (d) shows that the first-superdiagonal magnitude o_t stays near the Gaussian-reference value 0. Overall, these statistics provide empirical evidence that the accumulated real-image mismatch creates a persistent distributional gap between $\widehat{\boldsymbol{\varepsilon}}_t^{\text{real}}$ and $\widehat{\boldsymbol{\varepsilon}}_t^{\text{gen}}$. However, they do not directly show where the real and generated predicted noises are located in noise space. To better understand why $\boldsymbol{z}_T$ are more suitable for AIGI detection, we conduct a *local* analysis, which examines patchwise latent distributions and reveals spatially localized residual mismatch at late inversion timesteps.

4.3 Patchwise Latent-energy Distributions and Gaussian-prior Consistency Analysis

For a sample generated by the same diffusion model, let $\boldsymbol{\xi} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$ denote its source Gaussian seed. In the ideal inversion case, the terminal latent recovers this seed: $\boldsymbol{z}_T^{\text{gen}} = \boldsymbol{\xi}, \boldsymbol{\xi} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$. Hence the terminal latent of a generated sample is fully consistent with the Gaussian prior in theory. For a real image, the terminal latent contains the accumulated mismatch term derived above: $\boldsymbol{e}_T = \boldsymbol{z}_T^{\text{real}} - \boldsymbol{z}_T^{\text{gen}}$, or equivalently $\boldsymbol{z}_T^{\text{real}} = \boldsymbol{z}_T^{\text{gen}} + \boldsymbol{e}_T$. Therefore, generated terminal latents are expected to remain closer to the Gaussian prior, whereas real terminal latents generally include a structured deviation induced by the accumulated mismatch terms. To study this effect, we divide each latent tensor into small spatial patches and define the *patchwise latent energy* of a patch $\mathcal{P}$ by $E(\mathcal{P}) := \sum_{i\in\mathcal{P}} z_i^2 = \|\boldsymbol{z}_{\mathcal{P}}\|_2^2$, where m is the patch dimension, *i.e.* the number of latent values in the patch. If the entries of a latent patch are approximately $i.i.d\,\mathcal{N}(0,1)$ variables, then its energy follows approximately a χ_m^2 distribution with degrees of freedom m. Hence, comparing patchwise latent-energy distributions to the corresponding χ_m^2 reference provides a local test of Gaussian-prior consistency.

Fig. 2 shows these patchwise latent-energy distributions at start and terminal DDIM timesteps. As inversion proceeds, the energy distributions of the generated latent patches move substantially closer to the χ_m^2 reference, consistent with the terminal latent becoming closer to a Gaussian prior. The

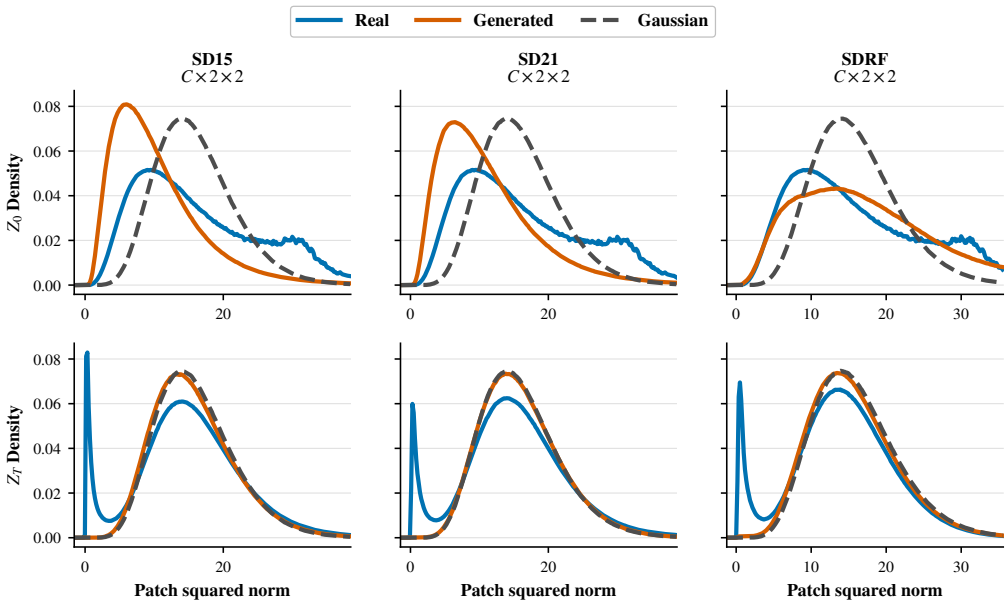


Figure 2: Patchwise latent-energy distributions of real, generated, and Gaussian-reference latents at $t = 0$ and $t = T$ for SD15, SD21, and SDRF.

real latent patches, however, remain less consistent with the χ^2_m reference at $t = T$, indicating that some local regions cannot be mapped back to Gaussian-like latent statistics by DDIM inversion. In particular, at $t = T$, the real curves show excess mass near very small squared norms and broader or more distorted shapes than the generated and Gaussian-reference curves (the blue peak in Fig. 2). Further, the analysis shows that this peak is not shared uniformly across all real images. When we remove images for which more than 10% of pixels take the same value, corresponding to approximately 38% of the real-image set, the low-energy peak largely disappears, and the remaining real-image patchwise distribution becomes much closer to the χ^2_m reference (see Appendix Fig. 6). This indicates that the pronounced low-energy peak is mainly driven by a subset of real images containing large plain or nearly constant regions or pixels. For this subset, the terminal latent $\boldsymbol{z}_T$ provides an especially strong classification signal, since these local regions fail to invert back to the χ^2_m behavior expected from Gaussian-like patches. More clearly, a simple Chi-squared test can serve as an effective detector for these images. At the same time, this low-energy peak does not explain the entire real-image set. Some samples of this peak are provided in Fig. 9 of the Appendix. The local analysis supports that one subset of real images is easy to classify because it retains obvious localized non-Gaussian residuals at $\boldsymbol{z}_T$, whereas the remaining cases require further analysis.

4.4 Quadratic Discriminant Analysis.

To further analyze the remaining real images, we study whether their terminal latent patches can also be approximated by Gaussian distributions. The theory of DDIM inversion suggests that $\boldsymbol{z}_T^{\text{gen}}$ should approximately follow $\mathcal{N}(\mathbf{0}, \boldsymbol{I})$. Moreover, the statistics in Fig. 1 and the approximately χ^2_m-like patch-energy distributions in Fig. 2, suggest that the latent patches of the remaining real images may also approximately follow the Gaussian. If both real and generated latent patches are approximated by Gaussian distributions, then *Quadratic Discriminant Analysis* (QDA), which is the optimal classifier under Gaussian class-conditional assumptions, should be able to classify them accurately.

As shown in Table 1, even *Quadratic Discriminant Analysis* (QDA), a simple classical Bayes-based classifier within the discriminant-analysis framework initiated by Fisher (1936) and extended to multivariate classification by Anderson (1951), achieves satisfactory performance by patch-level voting. For the full dataset without plain pixels, the $\boldsymbol{z}_T$ results suggest that the real and generated distributions can be approximated as two Gaussian distributions with different means and covariance matrices. To evaluate the full dataset, we extend the method to a hybrid detector based on patch-level voting. For each image, patches identified by the χ^2_m tail rule provide one detection signal, while

the remaining patches are classified by QDA. These patch-level decisions are then aggregated into a single image-level prediction. For reference, we also report $\boldsymbol{z}_0$ results. Since the $\boldsymbol{z}_0$ distribution is not close to Gaussian, its performance is substantially worse. Details of experiments can be found in Appendix E. Together, these results show that the advantage of using $\boldsymbol{z}_T$ is not only stronger separation, but also a simplified latent distribution in which even simple classifiers are effective.

Overall, DDIM inversion improves AIGI detection by increasing real–generated separation while producing a simpler, more prior-aligned distribution, motivating $\boldsymbol{z}_T$ as the detection representation.

Table 1: QDA classification accuracy (%) across patch sizes for SD15, SD21 and SDRF at $\boldsymbol{z}_0$ and $\boldsymbol{z}_T$. **Full w/o Plain:** QDA accuracy on the dataset without plain pixels. **Full:** results obtained by combining low-energy peak detection based on a 0.01% left-tail χ^2_m threshold with QDA on the non-plain subset.

	$\boldsymbol{z}_0$						$\boldsymbol{z}_T$					
	Full w/o Plain			Full			Full w/o Plain			Full		
Patch Size	SD15	SD21	SDRF	SD15	SD21	SDRF	SD15	SD21	SDRF	SD15	SD21	SDRF
$C \times 2 \times 2$	**72.42**	**71.37**	80.48	**79.90**	**77.35**	76.10	91.05	91.94	85.48	94.32	91.03	83.99
$C \times 4 \times 4$	57.26	62.26	76.85	59.77	62.57	71.61	95.56	92.90	92.02	97.25	**92.65**	88.93
$C \times 8 \times 8$	51.77	56.94	79.68	53.18	56.40	74.66	**97.34**	**93.15**	95.16	**98.38**	88.67	92.65
$C \times 16 \times 16$	50.32	53.39	**81.29**	49.94	53.47	**77.32**	96.85	92.26	**95.40**	98.27	72.70	**93.77**

4.5 Inversion Noise Classification

The above analysis motivates using the terminal latent $\boldsymbol{z}_T$ directly for AIGI detection. While individual patches of $\boldsymbol{z}_T$ are approximately Gaussian, the image-level latent retains structured, non-Gaussian dependencies that are difficult to model explicitly with limited data. We propose *Inversion Noise Classification* (INC): given an image $\boldsymbol{x}$, we encode it as $\boldsymbol{z}_0 = Enc(\boldsymbol{x})$, apply deterministic DDIM inversion to obtain $\boldsymbol{z}_T$, and directly classify it with ResNet-18. We evaluate INC on both biased and bias-controlled data for fair comparisons. QDA already performs strongly, indicating that low-order statistics capture much of the real–fake separation, while ResNet-18 can further exploit higher-order and cross-patch structure. Unlike reconstruction-based methods, INC requires neither resampling nor reconstruction-error computation. The algorithm is provided in Appendix G.

5 Experiments

5.1 Experimental Setup

Benchmarks. We benchmark INC against six reconstruction-based AIGI detectors: DIRE (Wang et al., 2023), SeDID (Ma et al., 2023), Aeroblade (Ricker et al., 2024), FIRE (Chu et al., 2025), FakeInversion (Cazenavette et al., 2024), and DTAD (Liang et al., 2025). DIRE classifies image reconstruction error $|\boldsymbol{x}' - \boldsymbol{x}|$, SeDID uses reconstruction error at an intermediate sampling step $\boldsymbol{z}'_t - \boldsymbol{z}_t$, and Aeroblade thresholds LPIPS (Zhang et al., 2018) VAE reconstruction error. FIRE uses frequency-guided reconstruction error, FakeInversion uses features from diffusion inversion and reconstruction, and DTAD measures image similarity along the denoising trajectory. Together, they provide representative, diverse baselines for direct comparison with INC.

Models. We study AIGI detection for large-scale LDMs accessed through the Hugging Face library (Hugging Face, 2026). We consider nine models spanning a range of architectures and prediction parameterizations. They are used for both text-to-image (T2I) generation and image-to-image (I2I) latent extraction. Details are listed in Table 10 in Appendix I.

Datasets. We conduct our analysis on a set of 10,000 images randomly sampled from LAION-Aesthetics v2 4.75 (Schuhmann et al., 2022), which we denote as "LAION-10k".[1] LAIONAesthetics is a large-scale, diverse collection of high-quality real-world images. We also conduct our experiment

[1]LAION provides image URLs and captions rather than the image files themselves. Therefore, the images used in our experiments remain subject to the licenses and terms of their original sources, and are used only for academic research.

on ImageNet-10k (Deng et al., 2009), Generated images are produced via T2I using the captions associated with the images, ensuring that the generative prompts are drawn from the same real-world distributions. Details on the T2I generation and I2I latent extractions are described in Appendix I.

Classifiers. For fair comparison, we utilize ResNet-18 (He et al., 2015) as the classifier architecture for all classifier-based benchmark methods. Details on classifier implementation are given in Appendix J.

5.2 RESULTS

We evaluate INC against diffusion-based AIGI detectors across variations in training/test T2I generators, I2I inversion models, image distortions and compression, and classifiers, assessing accuracy, robustness, and cross-model generalization. Unless otherwise specified, **bold** and underline indicate the best and second-best results, respectively, while ★ marks the overall highest value.

Table 2 benchmarks the results on SD15-generated images using different I2I models. This setting isolates the effect of the inversion model while keeping the T2I generator fixed, allowing us to assess different diffusion inversion pipelines. In practice, the I2I model can be selected by the method. INC achieves consistently high accuracy across I2I models, reaching 100% with SD35 and FLUX and performing overall the best. FakeInversion is also consistently strong, while DIRE, SeDID, FIRE, and $\boldsymbol{z}_0$ show greater variation across I2I models. Aeroblade and DTAD degrade more substantially under cross-model settings, indicating greater sensitivity to the inversion model. Overall, the results show that INC remains reliable when the I2I model differs from the T2I generators.

Table 2: Benchmarking classification accuracy (%) of various I2I methods on a dataset containing real and SD15-generated images.

I2I Method	SD15	SDRF	SD21	SD-T	SD35	SDXL	XL-T	PixA	FLUX	MAX
DIRE (Wang et al., 2023)	98.45	96.25	**98.55**	94.35	94.83	96.92	92.50	98.50	94.38	**98.55**
SeDID (Ma et al., 2023)	88.78	92.58	88.98	94.25	90.33	90.00	**96.63**	86.05	85.13	**96.63**
AEROBLADE (Ricker et al., 2024)	78.93	**81.26**	55.72	54.91	59.43	54.99	55.32	58.43	54.09	**81.26**
FIRE (Chu et al., 2025)	87.02	88.88	85.58	94.28	92.60	92.93	97.52	**99.45**	91.47	**99.45**
FakeInv (Cazenavette et al., 2024)	**99.88**	99.60	99.45	98.17	99.85	97.95	96.22	98.60	**99.88**	**99.88**
DTAD (Liang et al., 2025)	**93.44**	84.19	88.59	72.52	86.60	81.30	75.82	62.05	73.19	**93.44**
z_0	91.83	97.22	91.72	95.22	**98.33**	92.22	92.67	92.78	96.15	**98.33**
INC	99.85	98.88	98.92	98.85	**100.00**	97.33	94.40	97.10	**100.00**	**100.00**★

Complementing the previous cross-I2I evaluation, we fix SD15 as the I2I model and vary the T2I generator to evaluate cross-generator generalization. As shown in Table 3, INC consistently outperforms DIRE across all T2I methods, with accuracies ranging from 96.88% to 100% and a mean of 98.72%. INC also remains competitive with the strongest baselines across diverse generators, demonstrating robust generalization when the test-time generative model is unknown. Results after removing dataset bias, along with additional results on ImageNet-10k, are provided in Appendix K.

We further evaluate INC under mismatched train-test distortions from two perspectives: *robustness* to unseen test-time distortions and *generalizability* from distorted training data to clean test data. The same T2I and I2I model is used for training and testing, and details are provided in Appendix I.

Table 3: Benchmarking classification accuracy (%) using SD15 as the I2I model on datasets of real and generated images, where the generated images are produced by the specified T2I methods.

T2I Method	SD15	SDRF	SD21	SD-T	SD35	SDXL	XL-T	PixA	FLUX	Mean
DIRE (Wang et al., 2023)	98.45	98.95	96.60	98.28	95.75	93.95	92.50	95.00	92.35	95.76
SeDID (Ma et al., 2023)	99.03	99.67	96.47	**100.00**	98.20	96.53	**100.00**	97.62	96.95	98.27
AEROBLADE (Ricker et al., 2024)	79.30	82.17	58.50	59.58	55.38	58.73	60.68	53.30	57.67	62.81
FIRE (Chu et al., 2025)	87.02	92.55	83.90	97.55	91.67	90.32	98.22	93.47	86.82	91.28
FakeInv (Cazenavette et al., 2024)	**99.88**	**99.90**	**99.62**	99.95	**99.22**	**96.88**	**100.00**	**98.92**	**98.58**	**99.22**
DTAD (Liang et al., 2025)	93.44	93.45	92.68	93.42	93.26	87.77	93.24	88.11	79.76	90.57
z_0	93.03	97.38	91.42	**100.00**	97.22	94.83	99.88	95.45	92.88	95.79
INC	99.85	99.42	98.92	**100.00**	98.75	**96.88**	**100.00**	97.15	97.53	98.72

Table 4: Distortion robustness ablation (%). Classifiers are trained on undistorted data and tested on distorted data. The T2I and I2I methods utilize the same diffusion method indicated in the header.

Method →	DIRE				INC			
Distortion ↓	SD15	SDRF	SD21	SD-T	SD15	SDRF	SD21	SD-T
None	99.00	97.89	98.64	98.41	99.91	99.66	99.05	**100**
BDR-6	65.88	79.11	54.49	**84.16**	60.85	76.62	84.14	60.37
Blur-2.0	53.24	67.20	52.12	67.32	52.93	87.29	50.36	**97.97**
Grayscale	94.65	84.92	92.99	96.21	98.31	88.22	95.42	**99.97**
JPEG-25	57.78	75.50	55.50	60.49	67.16	69.48	**76.21**	53.98
Resize-25	67.61	84.71	68.76	60.46	78.90	94.55	81.36	**98.43**
Regen	78.92	88.96	78.80	97.95	67.26	76.18	95.27	**98.90**
Mean	69.68	80.07	67.11	77.77	70.90	82.06	80.46	**84.94**

For robustness, classifiers are trained on clean images and evaluated on distorted images. As shown in Table 4, INC is more robust than DIRE overall, achieving higher mean accuracy for all four diffusion-model settings and outperforming DIRE on most distortions. The gains are particularly pronounced for blur, resizing, grayscale, and regeneration, although both methods remain sensitive to severe degradations such as BDR and JPEG compression.

For generalizability, classifiers are trained on distorted images and tested on clean images. Table 5 shows that INC achieves the highest mean accuracy across all four settings, outperforming FakeInversion and DIRE by up to 28.71% and 18.40% points, respectively. INC remains effective even when trained with strong distortions such as JPEG compression and resizing, whereas reconstruction-based methods may exploit less reliable reconstruction cues as class signifiers, limiting their performances.

Finally, we ablate the inversion timestep used for classification. As shown in Fig. 7 of Appendix K, accuracy generally increases from (t=0) to (t=T), further supporting our hypothesis that higher-timestep inversion latents are more amenable to AIGI classification.

Table 5: Distortion generalization ablation (%). Classifiers are trained on distorted data and tested on undistorted data. The T2I and I2I methods utilize the same diffusion method indicated in the header.

I2I / T2I →	SD15			SDRF			SD21			SDT		
Method	DIRE	FakeInv	INC	DIRE	FakeInv	INC	DIRE	FakeInv	INC	DIRE	FakeInv	INC
BDR2	50.18	55.34	**73.74**	50.16	57.81	**79.37**	50.01	52.12	**56.93**	50.09	96.16	**98.60**
Blur2	49.97	51.93	**54.83**	57.50	71.23	**86.45**	50.00	**50.99**	50.06	50.02	**99.86**	99.78
Grayscale	72.44	73.02	**95.42**	94.21	**94.33**	83.42	**87.66**	63.12	73.54	94.32	94.23	**98.46**
JPEG25	50.01	50.29	**81.44**	52.55	59.69	**84.35**	50.00	50.84	**58.15**	50.32	65.22	**98.74**
Resize25	72.27	**99.72**	98.75	61.17	73.07	**90.06**	64.99	67.93	**84.22**	67.99	**99.90**	99.62
Renoise	**93.59**	77.20	91.78	95.71	**98.77**	98.04	**88.87**	86.32	82.39	**84.77**	74.57	74.56
Average	64.74	67.92	**82.66**	68.55	75.82	**86.95**	65.25	61.89	**67.55**	66.25	88.32	**94.96**

6 Conclusion

This work studies AIGI detection through deterministic DDIM inversion. We show that inversion errors accumulate recursively along the latent trajectory, and that real images contain an additional mismatch term that prevents them from converging to the same terminal distribution as generated images. This explains why late inversion timesteps are especially informative for detection. Our analysis further shows that DDIM inversion transforms the original complex latent space into a simpler and more prior-aligned terminal space $\boldsymbol{z}_T$, where generated samples move closer to the Gaussian prior, while some real samples cannot be fully inverted back and others remain systematically shifted. Motivated by these findings, we propose Inversion Noise Classification (INC), which detects AIGI directly in $\boldsymbol{z}_T$. Experiments across multiple diffusion models show that this approach is effective, robust, and more generalizable than the compared state-of-the-art methods.

AI USE STATEMENT

In this work, we have not used generative AI tools to help develop theoretical models or conceptual frameworks, formulate mathematical claims, provide critical ingredients for proving mathematical claims, assist in the writing of proofs, propose or refine hypotheses, design or provide feedback on research methodology or experiments, implement methods, support qualitative and thematic data analysis, interpret results, and generate synthetic data sets, assist with translation, clean and reformat datasets, or support qualitative and thematic data analysis do not apply to this work. Additionally, we used generative AI tools for sourcing/searching for information, editing a research paper to improve readability, and identifying relevant literature. We have reviewed all AI-assisted work. We have verified the authenticity of the provided information and the relevant literature. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

We have read and adhere to the ICLR Code of Ethics. This work involves no human subjects and no newly collected personal data. Real images come from LAION-Aesthetics and ImageNet, which remain subject to the licenses and terms of their original sources and are used solely for academic research. Generated images are produced with publicly available models under their respective licenses (Table 10). Risks of misuse, including erroneous decisions and adversarial adaptation, are discussed in Appendix M. We did not evaluate performance across demographic groups or content categories, so disparities cannot be ruled out.

REPRODUCIBILITY STATEMENT

To support reproducibility, we provide the assumptions and complete proof of Proposition 4.1 in Appendix A, and the INC algorithm in Appendix G. Data processing, T2I/I2I generation settings, and image distortions are described in Appendix I, classifier architecture and training in Appendix J, baseline implementations in Appendix H, and the statistical and QDA analysis protocols in Appendices C to E. We will release our code on GitHub upon acceptance.

REFERENCES

Inzamamul Alam, Muhammad Shahid Muneer, and Simon S Woo. Ugad: Universal generative ai detector utilizing frequency fingerprints. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pp. 4332–4340, 2024.

Mohd Ruhul Ameen and Akif Islam. Detecting ai-generated images via diffusion snap-back reconstruction: A forensic approach, 2026. URL https://arxiv.org/abs/2511.00352.

Theodore W Anderson. Classification by multivariate analysis. *Psychometrika*, 16(1):31–50, 1951.

Sanju N Attoor and Edward R Dougherty. Classifier performance as a function of distributional complexity. *Pattern Recognition*, 37(8):1641–1651, 2004.

Nora Belrose, Quintin Pope, Lucia Quirke, Alex Mallen, and Xiaoli Fern. Neural networks learn statistics of increasing complexity. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2024.

Giulio Biroli, Tony Bonnaire, Valentin De Bortoli, and Marc Mézard. Dynamical regimes of diffusion models. *Nature Communications*, 15(1):9957, 2024.

Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 18392–18402, 2023.

L. Burke-Preputnik, J. Dickson, A. Senko, and A. Friedman. Cyber scamming goes global: Sourcing forced labor for fraud factories. https://www.csis.org/analysis/cyber-scamming-goes-global-sourcing-forced-labor-fraud-factories. Accessed: 2026-04-15.

Junyoung Byun, Kyujin Shim, and Changick Kim. Bitnet: Learning-based bit-depth expansion. In C. V. Jawahar, Hongdong Li, Greg Mori, and Konrad Schindler (eds.), *Computer Vision – ACCV 2018*, pp. 67–82, Cham, 2019. Springer International Publishing. ISBN 978-3-030-20890-5.

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. IEEE, 2024.

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, 2024.

Beilin Chu, Xuan Xu, Xin Wang, Yufei Zhang, Weike You, and Linna Zhou. Fire: Robust detection of diffusion-generated images via frequency-guided reconstruction error. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12830–12839, 2025. doi: 10.1109/CVPR52734.2025.01197.

J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.

Prafulla Dhariwal and Alex Nichol. Diffusion models beat gans on image synthesis. In *Proceedings of the 35th International Conference on Neural Information Processing Systems*, NIPS '21, Red Hook, NY, USA, 2021. Curran Associates Inc. ISBN 9781713845393.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, Dustin Podell, Tim Dockhorn, Zion English, and Robin Rombach. Scaling rectified flow transformers for high-resolution image synthesis. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2024.

Ronald A Fisher. The use of multiple measurements in taxonomic problems. *Annals of eugenics*, 7 (2):179–188, 1936.

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

Jerome H Friedman. On bias, variance, 0/1—loss, and the curse-of-dimensionality. *Data mining and knowledge discovery*, 1(1):55–77, 1997.

Daniel Garibi, Or Patashnik, Andrey Voynov, Hadar Averbuch-Elor, and Daniel Cohen-Or. Renoise: Real image inversion through iterative noising. In *European Conference on Computer Vision*, pp. 395–413. Springer, 2024.

T. Germain. Your phone edits all your photos with ai - is it changing your view of reality? https://www.bbc.com/future/article/20260203-the-ai-that-quietly-edits-all-of-your-photos. Accessed: 2026-04-15.

Sandesh Ghimire, Jinyang Liu, Armand Comas, Davin Hill, Aria Masoomi, Octavia Camps, and Jennifer Dy. Geometry of score based generative models, 2023. URL https://arxiv.org/abs/2302.04411.

Kaiming He, X. Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2015. URL https://api.semanticscholar.org/CorpusID:206594692.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021. URL https://openreview.net/forum?id=qw8AKxfYbI.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA, 2020. Curran Associates Inc. ISBN 9781713829546.

Seongmin Hong, Kyeonghyun Lee, Suh Yoon Jeon, Hyewon Bae, and Se Young Chun. On exact inversion of dpm-solvers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7069–7078, 2024.

Tai-Ming Huang, Wei-Tung Lin, Kai-Lung Hua, Wen-Huang Cheng, Junichi Yamagishi, and Jun-Cheng Chen. Thinkfake: Reasoning in multimodal large language models for ai-generated image detection, 2025. URL https://arxiv.org/abs/2509.19841.

Hugging Face. Hugging face. https://huggingface.co/, 2026. Accessed: 2026-04-15.

Christopher Jarzynski. Nonequilibrium equality for free energy differences. *Physical review letters*, 78(14):2690, 1997a.

Christopher Jarzynski. Equilibrium free-energy differences from nonequilibrium measurements: A master-equation approach. *Physical Review E*, 56(5):5018, 1997b.

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

Matthew Joslin, Xian Wang, and Shuang Hao. Double face: Leveraging user intelligence to characterize and recognize AI-synthesized faces. In *33rd USENIX Security Symposium (USENIX Security 24)*, pp. 1009–1026, Philadelphia, PA, 2024. USENIX Association. ISBN 978-1-939133-44-1. URL https://www.usenix.org/conference/usenixsecurity24/presentation/joslin.

Xuan Ju, Ailing Zeng, Yuxuan Bian, Shaoteng Liu, and Qiang Xu. Pnp inversion: Boosting diffusion-based editing with 3 lines of code. *International Conference on Learning Representations (ICLR)*, 2024.

Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=k7FuTOWMOc7.

Valentin Khrulkov, Gleb Ryzhakov, Andrei Chertkov, and Ivan Oseledets. Understanding DDPM latent codes through optimal transport. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=6PIrhAx1j4i.

Diederik P Kingma and Ruiqi Gao. Understanding diffusion objectives as the ELBO with simple data augmentation. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=NnMEadcdyD.

Diederik P. Kingma, Tim Salimans, Ben Poole, and Jonathan Ho. Variational diffusion models, 2023. URL https://arxiv.org/abs/2107.00630.

Black Forest Labs, Stephen Batifol, Andreas Blattmann, Frederic Boesel, Saksham Consul, Cyril Diagne, Tim Dockhorn, Jack English, Zion English, Patrick Esser, and et al. Flux.1 kontext: Flow matching for in-context image generation and editing in latent space, 2025. URL https://arxiv.org/abs/2506.15742.

Yangming Li and Mihaela van der Schaar. On error propagation of diffusion models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=RtAct1E2zS.

Yachao Liang, Min Yu, Gang Li, Jianguo Jiang, Fuqiang Du, Li Jingyuan, Lanchi Xie, Zhen Xu, and Weiqing Huang. Denoising trajectory biases for zero-shot AI-generated image detection. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=2h8vXbEufN.

Shanchuan Lin, Bingchen Liu, Jiashi Li, and Xiao Yang. Common diffusion noise schedules and sample steps are flawed. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pp. 5404–5411, 2024.

Qiang Liu. Rectified flow: A marginal preserving approach to optimal transport, 2022. URL https://arxiv.org/abs/2209.14577.

Xingchao Liu, Chengyue Gong, and qiang liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=XVjTT1nw5z.

Xingchao Liu, Xiwen Zhang, Jianzhu Ma, Jian Peng, and qiang liu. Instaflow: One step is enough for high-quality diffusion-based text-to-image generation. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=1k4yZbbDqX.

Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver: a fast ode solver for diffusion probabilistic model sampling in around 10 steps. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA, 2022. Curran Associates Inc. ISBN 9781713871088.

Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver++: Fast solver for guided sampling of diffusion probabilistic models. *Machine Intelligence Research*, 22 (4):730–751, 2025.

Yuhang Lu and Touradj Ebrahimi. Towards the detection of ai-synthesized human face images. In *2024 IEEE International Conference on Image Processing (ICIP)*, pp. 3778–3784. IEEE, 2024.

Simian Luo, Yiqin Tan, Longbo Huang, Jian Li, and Hang Zhao. Latent consistency models: Synthesizing high-resolution images with few-step inference, 2024a. URL https://openreview.net/forum?id=duBCwjb68o.

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*, pp. 17006–17015, 2024b.

Lorenzo Luzi, Helen Jenne, Carlos Ortiz Marrero, and Ryan Murray. Using skew to assess the quality of GAN-generated image features. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=Io3jDUC4DP.

RuiPeng Ma, Jinhao Duan, Fei Kong, Xiaoshuang Shi, and Kaidi Xu. Exposing the fake: Effective diffusion-generated images detection. In *The Second Workshop on New Frontiers in Adversarial Machine Learning*, 2023. URL https://openreview.net/forum?id=7R62e4Wgim.

Karttikeya Mangalam and Vinay Uday Prabhu. Do deep neural networks learn shallow learnable examples first? In *ICML 2019 Workshop on Identifying and Understanding Deep Learning Phenomena*, 2019. URL https://openreview.net/forum?id=HkxHv4rn24.

Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. SDEdit: Guided image synthesis and editing with stochastic differential equations. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=aBsCjcPu_tE.

Daiki Miyake, Akihiro Iohara, Yu Saito, and Toshiyuki Tanaka. Negative-prompt inversion: Fast image inversion for editing with text-guided diffusion models. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 2063–2072. IEEE, 2025.

Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Null-text inversion for editing real images using guided diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6038–6047, 2023.

Anders Giovanni Moller, Daniel M. Romero, David Jurgens, and Luca Maria Aiello. The impact of generative ai on social media: An experimental study, Feb 2026. URL https://www.nature.com/articles/s41598-026-40110-8.

Radford M. Neal. Annealed importance sampling. *Statistics and Computing*, 11:125–139, 1998. URL https://api.semanticscholar.org/CorpusID:11112994.

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.

Gaurav Parmar, Krishna Kumar Singh, Richard Zhang, Yijun Li, Jingwan Lu, and Jun-Yan Zhu. Zero-shot image-to-image translation. In *ACM SIGGRAPH 2023 conference proceedings*, pp. 1–11, 2023.

Adrien Pavão. We are not able to identify ai-generated images, 2025. URL https://arxiv.org/abs/2512.22236.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4195–4205, 2023.

Maria Refinetti, Alessandro Ingrosso, and Sebastian Goldt. Neural networks trained with sgd learn distributions of increasing complexity. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org, 2023.

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

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

Thomas Roca, Anthony Cintron-Roman, Jehu Torres, Marcelo Duarte, Pengce Wang, Kevin White, Amit Misra, and Juan Lavista-Ferres. How good are humans at detecting ai-generated images? learnings from an experiment, 2025. URL https://arxiv.org/abs/2507.18640.

Robin Rombach, A. Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10674–10685, 2021. URL https://api.semanticscholar.org/CorpusID:245335280.

Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=TIdIXIpzhoI.

Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade W Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa R 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 *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2022. URL https://openreview.net/forum?id=M3Y74vmsMcY.

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

Jascha Sohl-Dickstein, Eric A. Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *Proceedings of the 32nd International Conference on Machine Learning - Volume 37*, ICML'15, pp. 2256–2265. JMLR.org, 2015.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=St1giarCHLP.

Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019.

Yang Song, Jascha Narain Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *ArXiv*, abs/2011.13456, 2020. URL https://api.semanticscholar.org/CorpusID:227209335.

Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org, 2023.

Łukasz Staniszewski, Łukasz Kuciński, and Kamil Deja. There and back again: On the relation between noises, images, and their inversions in diffusion models, 2025. URL https://openreview.net/forum?id=Dgh5GXsW65.

Peize Sun, Yi Jiang, Shoufa Chen, Shilong Zhang, Bingyue Peng, Ping Luo, and Zehuan Yuan. Autoregressive model beats diffusion: Llama for scalable image generation. *arXiv preprint arXiv:2406.06525*, 2024.

Chuangchuang Tan, Jinglu Wang, Xiang Ming, Renshuai Tao, Yunchao Wei, Yao Zhao, and Yan Lu. Forenx: Towards explainable ai-generated image detection with multimodal large language models, 2025. URL https://arxiv.org/abs/2508.01402.

Office to Monitor United States Department of State and Combat Trafficking in Persons. The impact of artificial intelligence on trafficking in persons. https://www.state.gov/wp-content/uploads/2026/04/The-Impact-of-Artificial-Intelligence-Accessible-4.16.2026.pdf. Accessed: 2026-04-15.

Enrico Ventura, Beatrice Achilli, Gianluigi Silvestri, Carlo Lucibello, and Luca Ambrogioni. Manifolds, random matrices and spectral gaps: The geometric phases of generative diffusion. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=KlN00vQEY2.

Jiangshan Wang, Junfu Pu, Zhongang Qi, Jiayi Guo, Yue Ma, Nisha Huang, Yuxin Chen, Xiu Li, and Ying Shan. Taming rectified flow for inversion and editing. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=uDreZphNky.

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

Shifeng Xu, Yanzhu Liu, and Adams Wai-Kin Kong. Variance-reduction guidance: Sampling trajectory optimization for diffusion models. In *2025 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1–6. IEEE, 2025.

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.

John Zhi-Qin, John Xu, Luo Tao, Xiao Yanyang, and Ma Zheng. Frequency principle: Fourier analysis sheds light on deep neural networks. *Communications in Computational Physics*, 28(5): 1746–1767, 2020.

Nan Zhong, Mian Zou, Yiran Xu, Zhenxing Qian, Xinpeng Zhang, Baoyuan Wu, and Kede Ma. Self-supervised ai-generated image detection: A camera metadata perspective. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2026.

Ziyin Zhou, Yunpeng Luo, Yuanchen Wu, Ke Sun, Jiayi Ji, Ke Yan, Shouhong Ding, Xiaoshuai Sun, Yunsheng Wu, and Rongrong Ji. Aigi-holmes: Towards explainable and generalizable ai-generated image detection via multimodal large language models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 18746–18758, 2025.

Haim Zisman and Uri Shaham. Realstats: A rigorous real-only statistical framework for fake image detection. In *The 29th International Conference on Artificial Intelligence and Statistics*, 2026. URL https://openreview.net/forum?id=qvwsVoH24j.

A. Zukanovic. Deepfakes and biometrics: Bypassing security with synthetic identities. https://www.luc.edu/uxandbiometricslab-dev/news/blog/archive/deepfakesandbiometricsbypassingsecuritywithsyntheticidentities.shtml. Accessed: 2026-04-15.

A Proof of Proposition 4.1

Proof. Since the schedule is monotone, we have $\alpha_{t+1} < \alpha_t$, $\sigma_{t+1} > \sigma_t$. Therefore, $0 < \frac{\alpha_{t+1}}{\alpha_t} < 1$, which implies $1 + a_t = \frac{\alpha_{t+1}}{\alpha_t} \in (0, 1), \qquad a_t = \frac{\alpha_{t+1}}{\alpha_t} - 1 < 0$. Moreover, $b_t = \sigma_{t+1} - \frac{\alpha_{t+1}}{\alpha_t}\sigma_t > \sigma_t - \frac{\alpha_{t+1}}{\alpha_t}\sigma_t = \sigma_t \left(1 - \frac{\alpha_{t+1}}{\alpha_t}\right) > 0$. Hence, the previous gap is propagated with a contraction factor $1 + a_t \in (0, 1)$, while each new mismatch term enters with a positive coefficient b_t.

Next, subtract the generated and real DDIM updates in Eq. 5:

$$\begin{aligned} \boldsymbol{e}_{t+1} &= \boldsymbol{z}_{t+1}^{\text{real}} - \boldsymbol{z}_{t+1}^{\text{gen}} \\ &= \big((1 + a_t)\boldsymbol{z}_t^{\text{real}} + b_t\, \widehat{\boldsymbol{\varepsilon}}_t^{\text{real}}\big) - \big((1 + a_t)\boldsymbol{z}_t^{\text{gen}} + b_t\, \widehat{\boldsymbol{\varepsilon}}_t^{\text{gen}}\big) \\ &= (1 + a_t)\big(\boldsymbol{z}_t^{\text{real}} - \boldsymbol{z}_t^{\text{gen}}\big) + b_t\big(\widehat{\boldsymbol{\varepsilon}}_t^{\text{real}} - \widehat{\boldsymbol{\varepsilon}}_t^{\text{gen}}\big). \end{aligned}$$

By the definition of the latent gap in Eq. 6, $\boldsymbol{z}_t^{\text{real}} - \boldsymbol{z}_t^{\text{gen}} = \boldsymbol{e}_t$, and by the error model in Eq. 4, $\widehat{\boldsymbol{\varepsilon}}_t^{\text{real}} - \widehat{\boldsymbol{\varepsilon}}_t^{\text{gen}} = (\boldsymbol{\varepsilon}_t^\star + \boldsymbol{\delta}_t + \boldsymbol{\gamma}_t) - (\boldsymbol{\varepsilon}_t^\star + \boldsymbol{\delta}_t) = \boldsymbol{\gamma}_t$. Substituting these into the previous expression gives

$$\boldsymbol{e}_{t+1} = (1 + a_t)\boldsymbol{e}_t + b_t\boldsymbol{\gamma}_t,$$

which proves Eq. 7.

We now unroll the recursion. For the first two steps, $\boldsymbol{e}_1 = (1 + a_0)\boldsymbol{e}_0 + b_0\boldsymbol{\gamma}_0$, and

$$\boldsymbol{e}_2 = (1 + a_1)\boldsymbol{e}_1 + b_1\boldsymbol{\gamma}_1 = (1 + a_1)(1 + a_0)\boldsymbol{e}_0 + (1 + a_1)b_0\boldsymbol{\gamma}_0 + b_1\boldsymbol{\gamma}_1.$$

Continuing this expansion inductively yields

$$\boldsymbol{e}_T = \left(\prod_{j=0}^{T-1}(1 + a_j)\right)\boldsymbol{e}_0 + \sum_{s=0}^{T-1}\left(\prod_{j=s+1}^{T-1}(1 + a_j)\right) b_s\boldsymbol{\gamma}_s,$$

which is Eq. 8.

Therefore, the terminal gap $\boldsymbol{e}_T$ consists of the propagated initial gap $\boldsymbol{e}_0$ together with the accumulated contributions of all extra real-image mismatch terms $\boldsymbol{\gamma}_0, \boldsymbol{\gamma}_1, \ldots, \boldsymbol{\gamma}_{T-1}$ up to timestep T. □

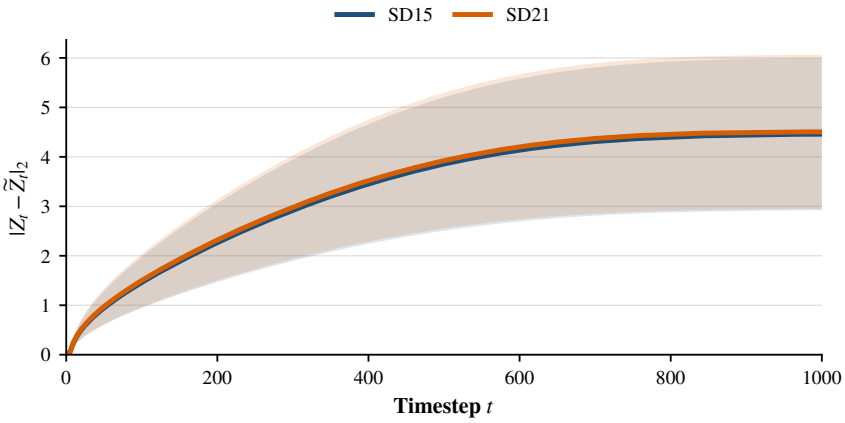


Figure 3: DDIM roundtrip latent discrepancy for generated samples across SD1.5 and SD2.1. Starting from the terminal latent $\boldsymbol{z}_T$, we reconstruct back to $\boldsymbol{z}_0$ and then invert again to obtain a reconstructed trajectory $\widetilde{\boldsymbol{z}}_t$. The plot shows the timestep-wise mean discrepancy $\|\boldsymbol{z}_t - \widetilde{\boldsymbol{z}}_t\|_2 \pm$ std (shaded region), which increases with t across both models, indicating that even the DDIM inversion error accumulates progressively along the trajectory.

B DDIM Roundtrip Discrepancy and Accumulated Noise-approximation Error

Figure 3 provides empirical evidence that the noise-approximation error accumulates along the DDIM inversion trajectory even in the generated case. Starting from the terminal latent $\boldsymbol{z}_T$, we reconstruct

back to $\boldsymbol{z}_0$ and then invert again to obtain a reconstructed trajectory $\widetilde{\boldsymbol{z}}_t$. The plotted discrepancy $\|\boldsymbol{z}_t - \widetilde{\boldsymbol{z}}_t\|_2$ increases steadily with t, showing that the inversion error becomes larger at later timesteps. This observation is consistent with Proposition 4.1: although the proposition focuses on the extra real-image mismatch term $\boldsymbol{\gamma}_t$, the same recursive DDIM structure also implies that the approximation error $\boldsymbol{\delta}_t$ is progressively accumulated along the trajectory. Therefore, even in the absence of the real-image-specific term, repeated inversion already introduces a growing discrepancy, while in the real-image case this effect is further amplified by the additional mismatch term $\boldsymbol{\gamma}_t$.

C Empirical Analysis of Inversion and Real-Image Mismatch Errors

We empirically examine the two error terms introduced in our analysis: the inversion approximation error δ_t and the additional real-image mismatch γ_t. Unless otherwise stated, all experiments use the LAION-10k dataset with $N = 10{,}000$ real images and $N = 10{,}000$ generated images. We report results for Stable Diffusion v1.5 (SD1.5) and Stable Diffusion v2.1 (SD2.1). For readability, the checkpoints are denoted by $t = 0, 100, \ldots, 1000$; these correspond to the discrete scheduler indices $0, 99, \ldots, 999$.

Definitions of δ_t and γ_t. For a generated image, the reference noise $\epsilon_t^\star$ is available from its generation trajectory. We invert the generated image through the same DDIM timesteps and obtain the predicted noise $\widehat{\epsilon}_t^{\mathrm{gen}}$. The inversion approximation error is defined as $\delta_t = \widehat{\epsilon}_t^{\mathrm{gen}} - \epsilon_t^\star$. This term captures the discrepancy caused by the fact that DDIM inversion does not exactly recover the original generation trajectory.

For real images, no reference generation trajectory or ground-truth noise $\epsilon_t^\star$ is available. Following Eq. 4, we assume that real and generated images inverted using the same diffusion model share the population-level inversion approximation component δ_t. Real images additionally exhibit a mismatch term γ_t because they do not originate from the generation manifold of the diffusion model. We therefore estimate its population mean as

$$\widehat{\mathbb{E}[\gamma_t]} = \frac{1}{N} \sum_{n=1}^{N} \left[\epsilon_\theta \left(z_{t,n}^{\mathrm{real}}, t \right) - \epsilon_\theta \left(z_{t,n}^{\mathrm{gen}}, t \right) \right],$$

and report its magnitude,

$$G_t = \left\| \widehat{\mathbb{E}[\gamma_t]} \right\|_2 .$$

Here, $z_{t,n}^{\mathrm{real}}$ and $z_{t,n}^{\mathrm{gen}}$ denote the inverted latent representations at timestep t, and ϵ_θ is the noise-prediction network.

Importantly, G_t is the norm of a population-mean vector estimated using all 10,000 samples, rather than the mean of per-sample scalar norms. It therefore produces one deterministic dataset-level value at each timestep, and a per-sample standard deviation is not directly defined for this statistic.

Persistence and accumulation of γ_t. We evaluate G_t together with the accumulated mismatch error

$$A_t^\gamma = \left\| \sum_{s=0}^{t} \left(\prod_{j=s+1}^{t} (1 + a_j) \right) b_s \, \widehat{\mathbb{E}[\gamma_s]} \right\|_2 .$$

Table 6 reports the results for SD1.5 and SD2.1. For both models, G_t remains nonzero at every evaluated timestep. Moreover, the accumulated mismatch increases smoothly from 0.0138 to 2.7619 for SD1.5 and from 0.0102 to 2.4468 for SD2.1. These observations indicate that γ_t persists throughout inversion and that its contribution accumulates progressively rather than arising as a sporadic numerical artifact.

Table 6: Estimated real-image mismatch and its accumulated error across inversion timesteps on LAION-10k.

Metric	0	100	200	300	400	500	600	700	800	900	1000
SD1.5 $\left\|\widehat{\mathbb{E}[\gamma_t]}\right\|_2$	1.0568	2.4594	2.6372	2.7103	2.7421	2.7620	2.7820	2.7948	2.8011	2.8192	2.8628
SD1.5 accumulated error	0.0138	0.6079	1.0556	1.4560	1.8049	2.0989	2.3297	2.4980	2.6148	2.6980	2.7619
SD2.1 $\left\|\widehat{\mathbb{E}[\gamma_t]}\right\|_2$	0.7591	2.0664	2.2169	2.2969	2.3493	2.3944	2.4366	2.4667	2.4874	2.5113	2.5425
SD2.1 accumulated error	0.0102	0.5106	0.8889	1.2350	1.5433	1.8042	2.0126	2.1749	2.2944	2.3807	2.4468

Direct measurement of the inversion approximation error. To verify the existence of δ_t, we generate 10,000 images from the LAION-10k captions using SD1.5 and record the reference noise $\epsilon_t^\star$ along each generation trajectory. We then invert the generated images using DDIM over the same timesteps and compute $D_{t,n} = \left\|\widehat{\epsilon}_{t,n}^{\text{gen}} - \epsilon_{t,n}^\star\right\|_2$. Unlike G_t, $D_{t,n}$ is a per-sample scalar; hence, its mean and standard deviation can be computed across the 10,000 samples.

As reported in Table 7, the mean approximation error generally increases with the inversion timestep, from 4.2459 at $t = 0$ to approximately 15 near the terminal timestep. This directly confirms that DDIM inversion introduces a nonzero approximation error that accumulates along the inversion trajectory.

Table 7: Mean and standard deviation of the per-sample inversion approximation error $\|\delta_t\|_2$ for SD1.5 on LAION-10k.

Statistic	0	100	200	300	400	500	600	700	800	900	1000
Mean	4.2459	5.9705	6.3518	6.9114	8.2089	9.9430	11.5310	13.2232	14.5780	15.4766	15.0261
Std.	2.9858	26.0005	21.9495	18.2018	18.0138	19.5213	20.7728	21.4926	21.7174	20.7070	19.5626

Relationship between accumulated mismatch and classification. Finally, we examine whether the accumulated mismatch is associated with the discriminative performance of z_t. Table 8 compares A_t^γ with the classification accuracy obtained from the corresponding latent representation.

For SD1.5, the Pearson correlation between accumulated mismatch and classification accuracy is $r = 0.7972, p = 0.0033$, while SD2.1 produces $r = 0.9266, p < 0.0001$. Thus, both models exhibit a statistically significant positive correlation. As the real-image mismatch accumulates, the latent representations become increasingly discriminative between real and generated images. The correlation is not perfectly linear because classification accuracy approaches saturation at later timesteps, particularly for SD1.5.

Table 8: Accumulated real-image mismatch and corresponding classification accuracy across inversion timesteps.

Metric	0	100	200	300	400	500	600	700	800	900	1000
SD1.5 accumulated error	0.0138	0.6079	1.0556	1.4560	1.8049	2.0989	2.3297	2.4980	2.6148	2.6980	2.7619
SD1.5 accuracy (%)	92.525	97.800	99.025	99.400	99.750	99.775	99.725	99.675	99.775	99.725	99.750
SD2.1 accumulated error	0.0102	0.5106	0.8889	1.2350	1.5433	1.8042	2.0126	2.1749	2.2944	2.3807	2.4468
SD2.1 accuracy (%)	91.625	91.875	95.100	96.650	97.900	98.025	97.850	98.200	98.025	97.950	98.150

Overall, these results support three conclusions. First, DDIM inversion introduces a measurable approximation error δ_t. Second, real images exhibit an additional population-level mismatch γ_t that remains nonzero and accumulates throughout inversion. Third, the accumulated γ_t error is strongly and significantly correlated with classification performance, supporting our explanation that inversion progressively amplifies the separation between real and generated images.

D KL DIVERGENCE, SKEWNESS, AND KURTOSIS.

As shown in Figure 2, both real and generated latents at z_0 remain far from Gaussian across all patch sizes. For smaller patches ($C \times 2 \times 2$ and $C \times 4 \times 4$), where the dimensionality is tractable, we adopt a *KNN-based KL estimator* to estimate KL divergence without imposing any distributional assumption. For larger patches ($C \times 8 \times 8$ and $C \times 16 \times 16$), however, KNN-based estimation becomes unreliable in high dimensions. As shown in Figure 4, the distributions at these scales are visually closer to Gaussian, so we instead adopt a *Gaussian KL* estimator. Since the Gaussian assumption does not hold at z_0, Gaussian KL is not applicable at this stage and is denoted "$-$" in Table 9.

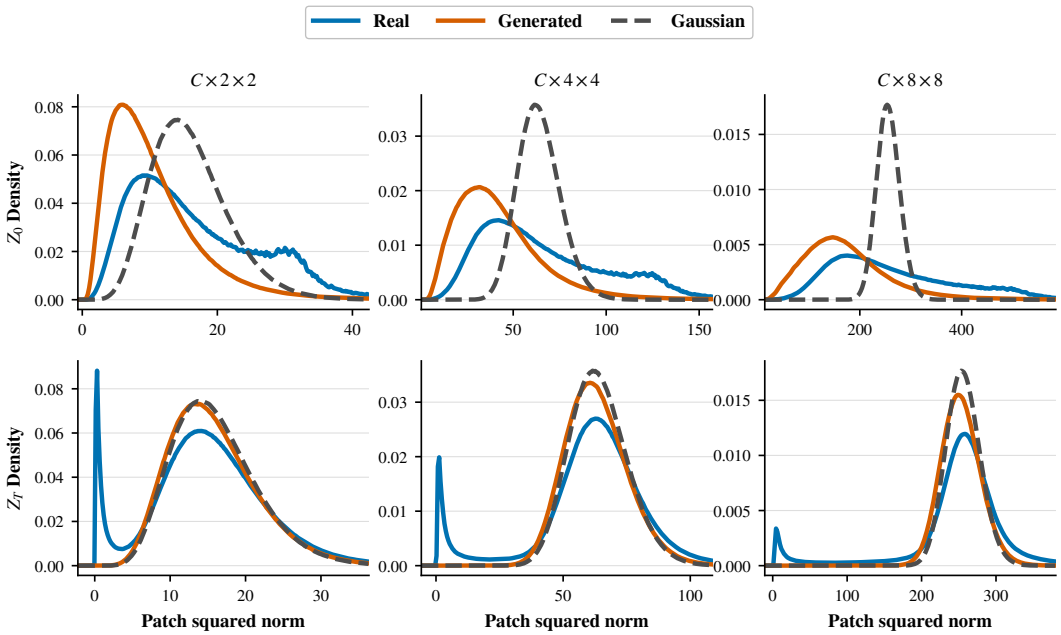


Figure 4: Patchwise latent-energy distributions of real, generated, and Gaussian-reference latents at $t = 0$ and $t = T$ for SD1.5 across patch sizes $C \times 2 \times 2$, $C \times 4 \times 4$, and $C \times 8 \times 8$.

After inversion, both real and generated latents become more Gaussian-aligned at z_T, which is also reflected in the *skewness* and *kurtosis* statistics: a Gaussian distribution is expected to have skewness close to 0 and kurtosis close to 3. This trend is consistent across both KL estimators. The KL values at z_T are markedly smaller than those at z_0, confirming that inversion substantially reduces the distributional gap from Gaussian. Among the two distributions, z_T^{gen} is numerically the closest to the Gaussian reference, as evidenced by the lower $\mathrm{G}\|\mathcal{N}$ values relative to $\mathrm{R}\|\mathcal{N}$. We report three KL values: $\mathrm{R}\|\mathcal{N}$ and $\mathrm{G}\|\mathcal{N}$ quantify how far each distribution deviates from the standard Gaussian, while $\mathrm{R}\|\mathrm{G}$ directly captures the distributional gap between real and generated latents. Since density estimation in high dimensions is unstable when data are limited, all distributional comparisons are conducted patchwise.

Table 9: Distribution statistics of patch-level features for SD1.5. KL divergence is estimated via KNN for smaller patches ($C \times 2 \times 2$, $C \times 4 \times 4$) and under a Gaussian assumption for larger patches; the latter is not applicable for $\mathbf{Z}_0$ (denoted $-$). $C \times H \times W$ denotes the full latent without spatial patching.

		Patch Size	KL			Skewness		Kurtosis	
			$\mathrm{R}\|\mathrm{G}$	$\mathrm{R}\|\mathcal{N}$	$\mathrm{G}\|\mathcal{N}$	$\hat{\gamma}_{\mathrm{R}}$	$\hat{\gamma}_{\mathrm{G}}$	$\hat{\kappa}_{\mathrm{R}}$	$\hat{\kappa}_{\mathrm{G}}$
$\mathbf{Z}_0$	KNN	$C \times 2 \times 2$	4.1726	12.5428	5.7981	-0.1584	-0.1033	3.364	4.099
	KNN	$C \times 4 \times 4$	13.9280	56.5291	29.7999	-0.1565	-0.1043	3.364	4.098
	Gaussian	$C \times 8 \times 8$	–	–	–	-0.1521	-0.1044	3.371	4.100
	Gaussian	$C \times 16 \times 16$	–	–	–	-0.1455	-0.1042	3.398	4.113
		$C \times H \times W^{\dagger\star}$	–	–	–	-0.1455	-0.0936	3.347	4.005
$\mathbf{Z}_T$	KNN	$C \times 2 \times 2$	1.6304	1.7940	0.0518	0.0059	0.0007	3.456	3.149
	KNN	$C \times 4 \times 4$	11.3848	12.3580	0.6311	0.0060	0.0002	3.435	3.149
	Gaussian	$C \times 8 \times 8$	0.3104	0.3578	0.2111	0.0064	-0.0005	3.397	3.149
	Gaussian	$C \times 16 \times 16$	1.2630	1.2590	0.9450	0.0071	-0.0015	3.342	3.149
		$C \times H \times W^{\dagger\star}$	2.99×10^9	4.36×10^4	4.36×10^4	0.0060	0.0011	3.477	3.149

†Full latent without spatial patching; features are the entire $C \times H \times W$ activation map flattened prior to distribution fitting.

*All statistics for the full-dimensional latent are for reference only; high dimensionality relative to available data points renders these estimates unreliable.

E Details of QDA

Compute Resources. Each patch-vote experiment required approximately 0.5–1.5 CPU-hours per model for SD15, SD21, and SDRF (with larger patch sizes requiring more time), executed on a 64-core AMD EPYC server with 251 GiB RAM. In practice, peak memory usage was approximately 32 GiB. No GPU was used in this experiment.

Dataset and Evaluation Protocol. We conduct QDA experiments on the LAION-10k dataset, which comprises 10,000 images. From this, we derive a filtered subset, denoted *Full w/o Plain*, by removing images in which more than 10% of pixels share an identical value. Such near-constant images are the primary source of the low-energy peak observed in the patch-wise latent energy distribution and are ill-suited for QDA. This filtering yields approximately 6,200 images. We partition *Full w/o Plain* into a training split (90%) and a held-out test set, Test Set 1 (10%); the removed plain images constitute a second evaluation set, Test Set 2.

Patch Extraction. Patches are extracted in overlapping fashion with a stride of 1, so each image of spatial dimension $h \times w$ yields $(h - p + 1)(w - p + 1)$ patches, where p denotes the patch size.

Inference and Voting. For each patch, QDA produces a score $\hat{p}_{\text{real}} \in [0, 1]$ denoting the posterior probability that the patch belongs to a real image. For the *Full w/o Plain* setting, we train QDA on the training split and evaluate on Test Set 1. The image-level score is computed as the mean $\hat{p}_{\text{real}}$ across all its patches. An image is classified as real if this mean exceeds 0.5, and as generated otherwise. For the full dataset setting (which includes plain images), we introduce a threshold-based correction to handle patches whose squared norm falls below the 0.01% left-tail quantile of the χ^2_m distribution. For such low-energy patches, we override the QDA score and assign $\hat{p}_{\text{real}} = 1$. All remaining patches retain their original QDA scores. The mean-score voting rule is then applied identically to the *Full w/o Plain* case. We report accuracy on both Test Set 1 and Test Set 2 to provide a comprehensive evaluation across both settings.

F PREDICTED-NOISE STATISTICS AFTER REMOVING THE PLAIN-PIXEL SUBSET

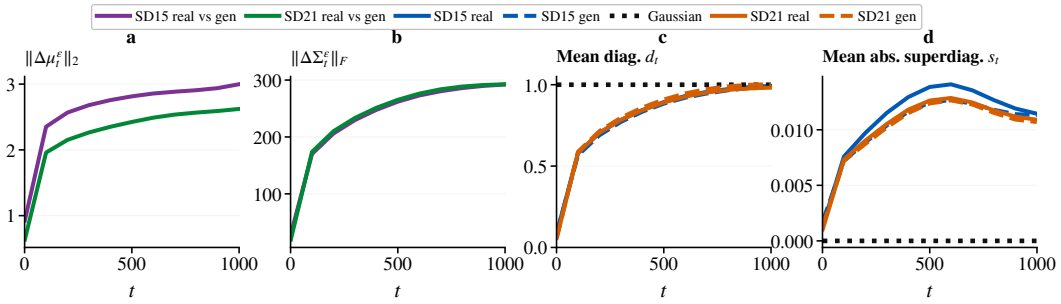


Figure 5: Predicted-noise distribution statistics after removing the plain-pixel subset. **(a)** $\|\Delta\boldsymbol{\mu}_t^\varepsilon\|_2$ and **(b)** $\|\Delta\Sigma_t^\varepsilon\|_F$ remain nonzero and increase before saturating, showing that a statistical gap between real and generated samples persists after filtering; compared with the full-dataset case, the gap is reduced. **(c)** Main-diagonal and **(d)** first-superdiagonal covariance statistics for SD1.5 and SD2.1. After filtering, the diagonal covariance curves become closer, while differences in the first-superdiagonal covariance remain.

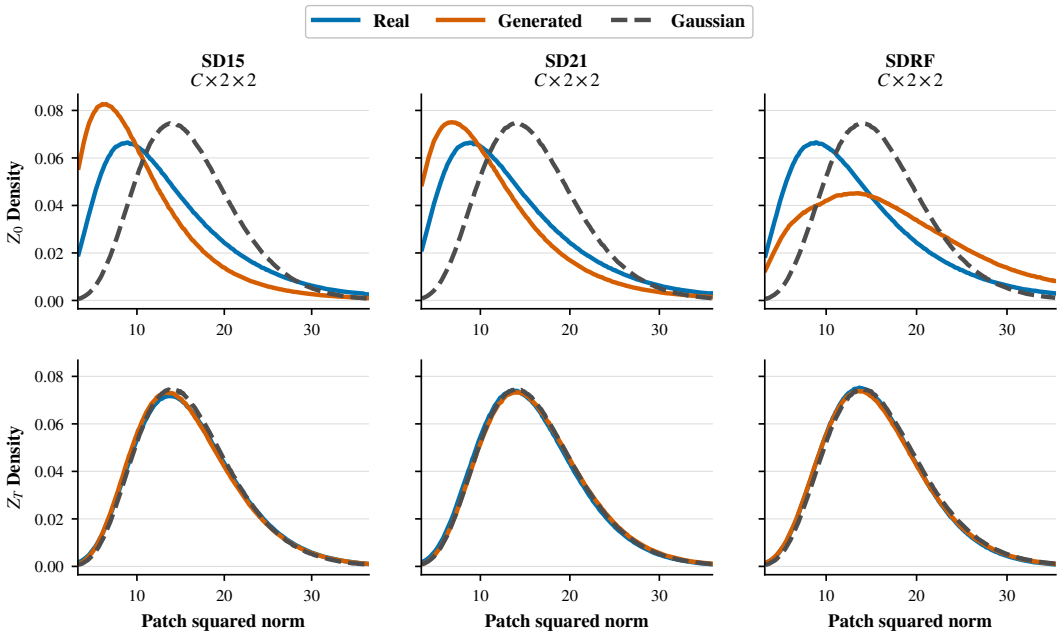


Figure 6: Patchwise latent-energy distributions after removing the plain-pixel subset, shown for real, generated, and Gaussian-reference latents at $t = 0$ and $t = T$ for SD1.5, SD2.1, and SDRF using $C \times 2 \times 2$ patches.

We examine the predicted-noise statistics after removing the plain-pixel subset. Figure 5 (a) and (b) show that both the predicted-noise mean gap $\|\Delta\boldsymbol{\mu}_t^\varepsilon\|_2$ and the covariance gap $\|\Delta\Sigma_t^\varepsilon\|_F$ remain nonzero and still increase with timestep before saturating at late timesteps. Compared with the full-dataset results in Figure 1, these gaps are reduced, indicating that the plain-pixel subset contributes substantially to the strongest real-vs-generated discrepancy. However, the separation does

not disappear after filtering, which shows that the remaining real and generated samples are still statistically distinguishable in predicted-noise space.

Figure 5 (c) and (d) further examine the covariance structure through the mean main-diagonal statistic and the mean absolute first-superdiagonal statistic. The diagonal curves for real and generated samples become much closer after removing the plain-pixel subset, while the first-superdiagonal curves still show a visible gap. Thus, after filtering, the remaining real and generated predicted-noise distributions are closer to each other than in the full dataset, but they are still not identical.

Figure 6 shows the patchwise latent-energy distributions after removing the plain-pixel subset. At $t = 0$, the patch-energy distributions of both real and generated images deviate substantially from the Gaussian reference, with model-dependent differences in their modes and shapes. After DDIM inversion to $t = T$, all distributions move markedly toward the Gaussian prior, and the gap between real and generated samples is greatly reduced. Nevertheless, small residual deviations remain, particularly for real-image latents, indicating that terminal latents are only approximately Gaussian at the patch level. This supports our observation that DDIM inversion produces a simpler, more prior-aligned representation while preserving subtle real–generated differences useful for detection.

G ALGORITHM

Algorithm 1 Inversion Noise Classification (INC)

Require: Real-image dataset X_{real}, generated-image dataset X_{gen}, VAE encoder $Enc(\cdot)$, DDIM inversion operator $\mathcal{I}_T(\cdot)$, classifier $f_\phi(\cdot)$
Ensure: Trained classifier f_ϕ

Initialize empty latent dataset $\mathcal{D} = \emptyset$
for each image $\boldsymbol{x} \in X_{\text{real}}$ **do**
  $\boldsymbol{z}_0 \leftarrow Enc(\boldsymbol{x})$
  $\boldsymbol{z}_T \leftarrow \mathcal{I}_T(\boldsymbol{z}_0)$
  $\mathcal{D} \leftarrow \mathcal{D} \cup \{(z_T, 0)\}$ ▷ label 0: real
end for
for each image $\boldsymbol{x} \in X_{\text{gen}}$ **do**
  $\boldsymbol{z}_0 \leftarrow Enc(\boldsymbol{x})$
  $\boldsymbol{z}_T \leftarrow \mathcal{I}_T(\boldsymbol{z}_0)$
  $\mathcal{D} \leftarrow \mathcal{D} \cup \{(z_T, 1)\}$ ▷ label 1: generated
end for
$\mathcal{D}_{\text{train}}, \mathcal{D}_{\text{test}} \leftarrow \text{Split}(\mathcal{D}, 0.8)$ ▷ 80/20 train-test random split
Train classifier f_ϕ on D_{train} using $\boldsymbol{z}_T$ as the input feature
Evaluate f_ϕ on D_{test}

H BENCHMARKS

We implement the DIRE method (Wang et al., 2023) following the algorithm described in their paper. Our I2I process settings for extracting $\boldsymbol{x}'$ and $\boldsymbol{x}$ for DIRE are identical to those of INC. SeDID (Ma et al., 2023) did not publish code and their description of their final timestep and step size in their paper is unclear. We tested multiple feasible configurations on a test dataset of real and SD15-T2I images and chose the implementation with the highest classification performance; our implementation follows SeDID by utilizing six inversion steps but differs in that reconstruction is analyzed at step 835 (*i.e.*, return $\boldsymbol{z}'_{835} - \boldsymbol{z}_{835}$). Lastly, Aeroblade is a training-free method, utilizing value thresholding to classify inputs instead of training a classifier network (Ricker et al., 2024). We utilized Aeroblade's published code and added a simple accuracy metric output.

Table 10: Diffusion models for T2I and I2I. Abbreviations: Latent Rectified Flow ⇒ LRF, Adversarial Diffusion Distillation ⇒ ADD, Guidance Distillation ⇒ GD.

Name	HuggingFace Path	Model Type	Prediction	License
SD15	stable-diffusion-v1-5/stable-diffusion-v1-5	LDM	Noise	CreativeML OpenRAIL-M
SDRF	XCLiu/2_rectified_flow-_from_sd_1_5	LRF-Reflow	Velocity	CC BY-NC 4.0
SD21	sd-research/stable-diffusion-2-1-base	LDM	Velocity	OpenRAIL++
SD-T	stabilityai/sd-turbo	LDM-ADD	Noise	Stability AI Community License
SD35	stabilityai/stable-diffusion-3.5-medium	LRF	Velocity	Stability AI AI Community License
SDXL	stabilityai/stable-diffusion-xl-base-1.0	LDM	Noise	OpenRAIL++
XL-T	stabilityai/sdxl-turbo	LDM-ADD	Noise	SAI-NC Community
PixA	PixArt-alpha/PixArt-Sigma-XL-2-1024-MS	LDM	Noise	OpenRAIL++
FLUX	black-forest-labs/FLUX.1-dev	LRF-GD	Velocity	FLUX.1-dev Non-Commercial License

I DATASETS

We perform T2I and I2I using nine popular diffusion models that bridge a wide expanse of architectures and prediction parameterizations, detailed in Table 10. To create sets of generated images, we utilize the 10,000 captions from LAION-10k and the T2I pipelines associated with each LDM. All models use their default solver type and order. For each T2I dataset, the number of sampling steps and CFG scale generally follow the recommendations of the model's original release and HuggingFace documentation. Output image sizes are the default for each model - 1024×1024 for SDXL, PixArt, and Flux, and 512×512 otherwise. To accommodate Flux's parameter count, we quantized the model to int8. T2I dataset pedigrees are detailed in Table 11.

We also generate I2I datasets for all real and generated images. I2I involves deterministically inverting each image to noise ($\boldsymbol{x} \rightarrow \boldsymbol{z}_0 \rightarrow \boldsymbol{z}_T$) and then deterministically denoising it ($\boldsymbol{z}_T \rightarrow \boldsymbol{z}'_0 \rightarrow \boldsymbol{x}'$). HuggingFace image-to-image pipelines are largely based on stochastic inversion, following the SDEdit process (Meng et al., 2022); a notable contribution of our work is the creation, unification, and standardization of deterministic inversion pipelines across multiple HuggingFace model types (including models beyond those studied in this paper) in our code library, to be released upon publication. We utilize conditional inversion and denoising. When processed by I2I, images are resized to the input size expected by each model We apply deterministic VAE encoding for each image. For each model, the step count and CFG scale are selected through an ablation study that measures reconstruction error ($||\boldsymbol{x}' - \boldsymbol{x}||_2$), noise normality ($\boldsymbol{z}_T$ mean and variance), and reconstruction visual inspection; the best-performing configuration per model is applied across all I2I datasets. I2I dataset pedigrees are detailed in Table 12.

Table 11: T2I configuration details.

Method	Solver	Order	Steps	Condition	CFG
SD15	DDIM	1	50	Caption	5.0
SDRF	Euler	1	50	Caption	5.0
SD21	DDIM	1	50	Caption	5.0
SD-T	Euler	1	1	Caption	1.0
SD35	FM-Euler	1	50	Caption	5.0
SDXL	Euler	1	40	Caption	7.0
XL-T	Euler	1	4	Caption	1.0
PixA	DPM	2	20	Caption	4.5
FLUX	FM-Euler	1	50	Caption	3.5

Table 12: I2I configuration details.

Method	Solver	Order	Steps	Condition	CFG Fwd	CFG Rev
SD15	DDIM	1	50	Caption	1.0	1.0
SDRF	Euler	1	50	Caption	1.0	1.0
SD21	DDIM	1	50	Caption	1.0	1.0
SD-T	Euler	1	8	Caption	1.0	1.0
SD35	FM-Euler	1	150	Caption	1.0	1.0
SDXL	Euler	1	40	Caption	1.0	1.0
XL-T	Euler	1	4	Caption	1.0	1.0
PixA	DPM	2	40	Caption	1.0	1.0
FLUX	FM-Euler	1	50	Caption	1.0	1.0

We additionally consider multiple methods of distortion and compression on real and generated images. Distortions are applied to generated images directly after the T2I generation process and before the I2I latent extraction process. Distortions include bit-depth reduction (reduce 8-bit pixel values to 2-bits) following the algorithm applied in (Byun et al., 2019), Gaussian blur ($\sigma = 2.0$), grayscale conversion, JPEG compression (25% quality), image resizing (reduce height and width by 50% before restoring to full size), and diffusion regeneration (apply I2I to real images and use the output as the new "real" image). For regeneration, we apply the same I2I method as the T2I and I2I methods used in generation.

J Classifiers and Implementation Details

In order to reduce variability across experiments, we apply the same classifier architecture and training scheme to all tests. We utilize ResNet-18 (He et al., 2015) as the classifier. Images are resized to 512×512 before classification. Given that latents have a smaller resolution (typically $4 \times 64 \times 64$), we apply the "CIFAR mod" to the ResNet when classifying latents (reduce the kernel size of the first convolutional layer from 7×7 to 3×3 and omit the first maxpool layer). Classifiers are trained for 50 epochs, with a batch size of 128, 8 gradient accumulation steps, 0.001 initial learning rate, cosine learning rate decay, and 0.0001 weight decay. Each dataset is composed of the I2I outputs from $10,000$ real and $10,000$ generated images, with a $80/20$ train/test split. For each training run, we save the classifier corresponding to the highest test accuracy. All experiments report test accuracy, unless stated otherwise. Classifier train and test sets have the same T2I / I2I pedigree (*i.e.*, they come from the same T2I / I2I processes), unless otherwise specified. Each classification experiment is finished on a single NVIDIA RTX A5000 GPU card with 24GB of memory in around 2 hours.

K More Results

K.1 Bias-Controlled Evaluation

To verify that the observed performance is not driven by the plain-pixel bias identified in the real-image dataset, we repeat the main experiments after removing images containing substantial plain regions. We evaluate both the cross-I2I setting, where the T2I generator is fixed, and the cross-T2I setting, where the I2I model is fixed.

Table 13 reports the cross-I2I results after removing the plain-pixel subset. INC retains a maximum accuracy of 100%, unchanged from the full dataset, while most competing methods exhibit only modest changes. This indicates that the strong performance of INC across inversion models is not attributable to the plain-pixel bias.

The same trend holds when fixing SD15 as the I2I model and varying the T2I generator. As shown in Table 14, INC achieves a mean accuracy of 98.25%, only 0.47 percentage points below the full-dataset result, and remains competitive with the strongest baseline. These results further confirm that the main cross-generator conclusions are preserved after bias removal.

Figure 9 further illustrates the removed bias: real images with large plain regions produce many near-zero-energy patches after inversion, unlike typical real and generated samples.

Table 13: Classification accuracy (%) on the Full w/o Plain subset, using various I2I models on real and SD15-T2I images. MAX reports the maximum value in each row, with the change relative to the Full dataset shown in parentheses.

I2I Method	SD15	SDRF	SD21	SD-T	SD35	SDXL	XL-T	PixA	FLUX	MAX
DIRE (Wang et al., 2023)	99.11	98.23	**99.56**	99.15	99.03	98.87	99.03	99.52	96.05	**99.56 (+1.01)**
SeDID (Ma et al., 2023)	98.99	99.19	98.75	98.43	**99.96**	93.15	92.30	91.33	**99.96**	**99.96 (+3.33)**
AEROBLADE (Ricker et al., 2024)	**89.52**	**89.52**	54.80	54.80	63.55	83.63	62.30	83.19	84.19	**89.52 (+8.26)**
FIRE (Chu et al., 2025)	88.19	83.91	81.13	92.98	88.59	91.65	96.77	**99.35**	89.15	**99.35 (-0.10)**
FakeInversion (Cazenavette et al., 2024)	99.68	99.44	99.31	98.87	99.92	99.27	96.69	98.47	**99.96**	**99.96 (+0.08)**
DTAD (Liang et al., 2025)	**94.53**	85.02	90.39	74.84	91.40	74.84	79.06	59.93	73.40	**94.53 (+1.09)**
z_0	88.83	89.80	89.31	89.27	**97.86**	89.72	90.04	88.47	94.96	**97.86 (-0.47)**
INC	99.60	99.40	99.44	98.27	99.96	95.81	91.45	93.23	**100.00**	**100.00 (0.00)**

Table 14: Classification accuracy (%) on the Full w/o Plain subset using SD15 as the I2I model across datasets generated by different T2I methods. Bold indicates the best result, and underline indicates the second-best result. Mean reports the row average, with the change relative to the Full dataset shown in parentheses.

T2I Method	SD15	SDRF	SD21	SD-T	SD35	SDXL	XL-T	PixA	FLUX	Mean
DIRE (Wang et al., 2023)	99.11	99.23	97.90	98.71	97.14	92.82	98.75	95.60	94.68	97.10 (+1.34)
SeDID (Ma et al., 2023)	98.99	99.76	93.95	**100.00**	98.10	94.44	99.96	97.50	97.10	97.76 (-0.51)
AEROBLADE (Ricker et al., 2024)	89.52	91.85	59.48	64.11	58.87	54.44	58.27	57.54	62.58	66.30 (+3.49)
FIRE (Chu et al., 2025)	88.19	93.51	81.09	93.99	90.08	81.77	97.90	90.85	85.73	89.23 (-2.05)
FakeInversion (Cazenavette et al., 2024)	**99.68**	99.72	**99.35**	**100.00**	**98.99**	95.12	**100.00**	**98.95**	**97.82**	**98.85 (-0.37)**
DTAD (Liang et al., 2025)	94.53	94.53	93.49	94.50	94.29	88.60	94.31	89.01	81.10	91.60 (+1.03)
z_0	88.83	96.45	88.51	**100.00**	96.69	92.42	99.80	95.48	91.90	94.45 (-1.34)
INC	99.60	**99.88**	97.06	**100.00**	98.19	**95.40**	**100.00**	97.06	97.02	98.25 (-0.47)

Comments: INC substantially outperforms FIRE and DTAD, while achieving performance close to FakeInversion with much lower computational cost:

- **Fewer NFEs:** INC requires only inversion, whereas FakeInversion performs both inversion and reconstruction, reducing NFEs by approximately 50%.
- **Smaller inputs:** INC uses a single latent-space input, while FakeInversion uses three pixel-space inputs, reducing storage and computation.
- **No online augmentation:** INC inputs can be precomputed and cached before classifier training.
- **Faster training:** Under the same setup (16k samples, batch size 32, 50 epochs, one GPU), FakeInversion takes about 6 min/epoch (5 h total), compared with 20 s/epoch (17 min total) for INC, yielding an approximately $18\times$ speedup for only a 0.50-percentage-point gap in mean accuracy.
- **Better generalization:** The simple inversion-noise representation also reduces reliance on distortion-specific shortcuts, improving generalization beyond the training distribution, as shown in Table 5.

K.2 Cross-Model and Classifier Generalization

K.2.1 Classifier Backbone Ablation

We replace ResNet-18 with ResNet-50 to test whether the observed performance depends on classifier capacity. Tables 15 and 16 evaluate the cross-I2I and cross-T2I settings, respectively.

Table 15: Classification accuracy (%) using ResNet-50 as the classifier backbone. Models are evaluated on real and SD15-generated images using different I2I methods. Bold indicates the best result in each row, and MAX reports the corresponding best performance.

Method	SD15	SDRF	SD21	SD-T	SD35	SDXL	XL-T	PixA	FLUX	MAX
SeDID (Ma et al., 2023)	99.33	99.58	99.22	98.88	99.90	95.80	95.67	94.62	**100.00**	**100.00**
DIRE (Wang et al., 2023)	99.10	98.50	99.47	99.42	98.60	**99.65**	99.08	99.60	98.17	**99.65**
FakeInv (Cazenavette et al., 2024)	99.90	99.40	99.40	98.92	**99.92**	99.22	99.28	98.83	99.85	**99.92**
z_0	92.30	92.38	91.83	92.28	**98.85**	93.62	93.50	93.78	97.15	**98.85**
INC	99.62	99.40	99.70	98.90	**100.00**	97.53	95.10	94.80	**100.00**	**100.00**

Table 16: Classification accuracy (%) using ResNet-50 across images produced by different T2I generators, with SD15 used as the I2I model. Bold indicates the best result in each column, and underline indicates the second-best result.

Method	SD15	SDRF	SD21	SD-T	SD35	SDXL	XL-T	PixA	FLUX	Mean
SeDID (Ma et al., 2023)	99.33	99.75	96.03	99.97	98.40	96.60	**100.00**	98.03	97.15	98.36
DIRE (Wang et al., 2023)	99.10	99.12	98.12	98.75	97.22	95.03	99.30	95.75	93.80	97.35
FakeInv (Cazenavette et al., 2024)	**99.90**	**99.83**	**99.55**	**100.00**	**99.20**	**97.38**	99.97	**98.72**	**98.30**	**99.21**
z_0	92.30	97.78	90.53	**100.00**	97.62	95.15	99.83	96.08	95.75	96.12
INC	99.62	**99.83**	96.90	**100.00**	98.90	96.67	**100.00**	97.03	97.08	98.45

Replacing ResNet-18 with ResNet-50 largely preserves the main performance trends, indicating that the results are not sensitive to classifier backbone choice. In the cross-I2I setting, INC reaches 100% maximum accuracy, matching SeDID and slightly exceeding FakeInv at 99.92%. When fixing SD15 as the I2I model and varying the T2I generator, FakeInv achieves the highest mean accuracy of 99.21%, followed by INC at 98.45%. INC achieves or shares the best performance on SDRF, SD-T, and XL-T, while remaining competitive across the remaining generators. Overall, the effectiveness of the terminal latent representation is largely insensitive to the choice between ResNet-18 and ResNet-50.

K.2.2 Joint T2I–I2I Generalization

We further evaluate INC under jointly mismatched T2I and I2I models. Because exhaustive evaluation is computationally expensive, Table 17 reports a representative subset of model combinations; its top row and first column correspond to the settings of Tables 2 and 3, respectively. INC maintains high accuracy across the evaluated permutations, demonstrating strong generalization to mismatched T2I and I2I backbones. SDRF, SD35, and FLUX provide particularly robust inversion representations, with minimum accuracies of 97.28%, 99.93%, and 99.70%, respectively. Images generated by distilled models such as SD-T and XL-T are also consistently easy to detect, achieving at least 99.85% accuracy across the evaluated I2I models. These results further support the robustness of INC in practical settings where the underlying generative model is unknown.

Table 17: INC performance across T2I / I2I permutations (%).

I2I Method → / T2I Method ↓	SD15	SDRF	SD21	SD-T	SD35	SDXL	XL-T	PixA	FLUX
SD15	99.85	98.88	98.92	98.85	100	97.33	94.40	97.10	100
SDRF	99.42	98.92	95.47	99.60	100	99.18	98.23	98.03	100
SD21	98.92	99.28	99.00	94.93	99.93	94.88	90.38	91.28	99.70
SD-T	100	100	100	100	100	99.95	99.88	99.85	100
SD35	98.75	98.10	98.63	98.08	-	-	-	-	-
SDXL	96.88	97.28	96.30	97.10	-	-	-	-	-
XL-T	100	100	100	100	-	-	-	-	-
PixA	97.15	98.08	96.73	96.95	-	-	-	-	-
FLUX	97.53	97.85	97.55	96.80	-	-	-	-	-

K.2.3 MIXED-I2I TRAINING

Table 18: Classification accuracy (%) when trained on Combine9 and tested on images generated by different T2I methods using SD15 as the I2I model. Bold indicates the best result, and underline indicates the second-best result.

Method	SD15	SDRF	SD21	SD-T	SD35	SDXL	XL-T	PixA	FLUX	Average
AEROBLADE (Ricker et al., 2024)	64.82	64.81	52.30	57.40	53.00	49.63	50.56	52.83	56.82	55.80
SeDID (Ma et al., 2023)	94.11	97.21	91.84	<u>98.44</u>	95.30	<u>94.58</u>	96.85	<u>95.31</u>	89.68	94.81
DIRE (Wang et al., 2023)	<u>97.50</u>	96.98	<u>92.81</u>	96.98	93.55	94.53	95.89	95.11	90.41	94.86
FakeInv (Cazenavette et al., 2024)	**98.65**	**98.33**	**97.36**	**98.91**	**96.78**	93.73	**98.36**	**95.78**	<u>92.33</u>	**96.69**
z_0	83.99	90.56	81.84	95.20	87.42	89.98	93.60	90.51	82.72	88.42
INC	97.08	<u>97.60</u>	92.61	98.39	<u>95.47</u>	**94.91**	<u>98.11</u>	94.53	**93.20**	<u>95.77</u>

We next evaluate whether a detector can learn from heterogeneous inversion representations using the Combine9 dataset. We randomly partition the 10,000 images across nine I2I methods, with approximately 1,111 samples processed by each method, while fixing the random seed so that the underlying image subsets remain matched across combinations. This design isolates generalization across T2I generators without introducing sample-selection differences. As shown in Table 18, INC achieves the second-highest average accuracy of 95.77%, behind FakeInv at 96.69%, and performs best on SDXL- and FLUX-generated images. These results show that INC remains effective when trained on heterogeneous latents produced by multiple inversion models rather than a single fixed I2I pipeline.

K.3 INVERSION TIMESTEP AND COMPUTATIONAL TRADE-OFF

K.3.1 EFFECT OF INVERSION TIMESTEP

We evaluate classification at intermediate inversion timesteps, where $t = 0$ corresponds to $\boldsymbol{z}_0$ classification and $t = T$ to INC. As shown in Fig. 7, accuracy generally increases with t, supporting our analysis that later inversion latents are more discriminative and more amenable to classification. An exception is SD-T, which already achieves high accuracy at $t = 0$, indicating that SD-T-generated images are readily distinguishable from real images even without inversion.

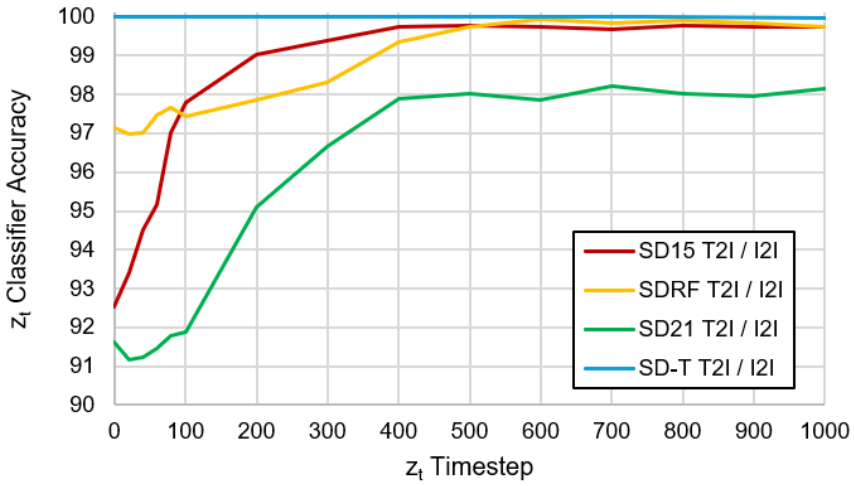


Figure 7: z_t classifier accuracy (%) across timesteps. T2I and I2I methods utilize the same diffusion model. Note that SD-T does not achieve 100% accuracy for z_0 in most T2I and I2I settings, such as those of Table 2.

K.3.2 Number of Inversion Steps

We compare INC using 10 and 50 inversion steps to quantify the accuracy–efficiency trade-off. Table 19 shows that reducing the inversion from 50 to 10 steps causes only modest average accuracy drops of 0.30 percentage points in the cross-I2I setting and 0.51 points in the cross-T2I setting. Thus, longer inversion provides the best accuracy, while substantially fewer steps remain a viable option when computational efficiency is prioritized.

Table 19: Runtime trade-off between 10 and 50 inversion steps. Top: SD15-generated images evaluated across different I2I inversion models. Bottom: images generated by different T2I models evaluated using SD15 inversion. Values in parentheses indicate the accuracy change relative to INC-50.

I2I Method	SD15	SDRF	SD21	SD-T	Mean
INC-50	99.67	99.55	99.60	98.89	99.43
INC-10	99.35 (-0.32)	99.47 (-0.08)	99.40 (-0.20)	98.30 (-0.59)	99.13 (-0.30)
T2I Method	**SD15**	**SDRF**	**SD21**	**SD-T**	**Mean**
INC-50	99.67	99.85	97.88	100.00	99.35
INC-10	99.35 (-0.32)	99.67 (-0.18)	96.40 (-1.48)	99.95 (-0.05)	98.84 (-0.51)

K.4 More Benchmark Dataset and Cross-Generator Generalization

K.4.1 ImageNet-10k

We repeat the main cross-I2I and cross-T2I evaluations on ImageNet-10k to test whether INC relies on LAION-specific cues. Table 20 shows that INC remains highly accurate across inversion models, reaching 100% accuracy with several I2I models. When SD15 inversion is fixed and the T2I generator varies, Table 21 shows that INC achieves a mean accuracy of 99.45%, second only to FakeInv at 99.66%. These results demonstrate strong transfer of the terminal-latent representation across datasets.

Table 20: Benchmarking classification accuracy (%) on ImageNet-10k using various I2I models on real and SD15-generated images. Bold indicates the best result in each row, and ★ marks the highest value in the MAX column.

I2I Method	SD15	SDRF	SD21	SD-T	SD35	SDXL	XL-T	PixA	FLUX	MAX
DIRE (Wang et al., 2023)	99.62	99.03	99.70	99.62	99.10	99.90	99.53	**99.95**	98.97	**99.95**
AEROBLADE (Ricker et al., 2024)	**95.50**	**95.50**	72.00	72.00	60.08	75.33	57.55	75.00	85.22	**95.50**
FIRE (Chu et al., 2025)	90.90	93.63	86.30	**97.55**	95.17	93.73	94.35	94.53	96.30	**97.55**
FakeInversion (Cazenavette et al., 2024)	99.95	99.72	99.85	99.55	99.95	99.85	99.45	99.85	**99.97**	**99.97**
DTAD (Liang et al., 2025)	87.91	86.64	81.38	58.58	87.47	85.05	64.15	**93.42**	80.81	**93.42**
z_0	97.58	97.35	97.10	97.58	99.20	**99.97**	95.12	99.95	**99.97**	**99.97**
INC	99.80	99.70	99.75	99.33	**100.00**	**100.00**	97.65	**100.00**	**100.00**	**100.00**★

Table 21: Benchmarking classification accuracy (%) on ImageNet-10k using SD15 as the I2I model on datasets of real and generated images, where the generated images are produced by the specified T2I methods. Bold indicates the best result, and underline indicates the second-best result.

T2I Method	SD15	SDRF	SD21	SD-T	SD35	SDXL	XL-T	PixA	FLUX	Mean
DIRE (Wang et al., 2023)	99.62	99.58	99.17	99.40	99.08	99.12	99.45	98.97	99.17	99.28
AEROBLADE (Ricker et al., 2024)	95.50	95.62	77.62	83.03	80.08	80.65	79.22	83.12	87.92	84.75
FIRE (Chu et al., 2025)	90.90	96.57	91.25	97.55	95.17	94.35	98.70	97.30	97.45	95.47
FakeInversion (Cazenavette et al., 2024)	**99.95**	**99.95**	**99.62**	**100.00**	**99.60**	98.80	**100.00**	**99.47**	**99.55**	**99.66**
DTAD (Liang et al., 2025)	87.91	87.91	87.72	87.90	87.85	87.33	87.89	85.54	85.96	87.33
z_0	97.58	98.75	96.60	**100.00**	98.67	98.70	99.92	98.65	99.35	98.69
INC	99.80	99.92	98.50	**100.00**	99.25	**99.15**	**100.00**	99.05	99.35	99.45

K.4.2 Autoregressive Generators

Although our theoretical analysis focuses on diffusion and flow models, we also evaluate empirical transfer to autoregressive generation using LlamaGen-XL (Sun et al., 2024). Images are generated from the same LAION-10k prompts and evaluated against real images under our standard protocol. As shown in Table 22, INC achieves 99.80%–99.92% accuracy across all tested I2I models and outperforms the compared methods in every setting, demonstrating strong transfer of the terminal-latent representation to autoregressive-generated images.

Table 22: Classification accuracy (%) on autoregressive-generated images from LAION-10k. Bold indicates the best result, and underline indicates the second-best result.

Method	SD15	SDRF	SD21	SDT
AEROBLADE (Ricker et al., 2024)	66.35	66.35	66.00	66.00
DIRE (Wang et al., 2023)	99.50	99.38	99.50	98.75
FakeInv (Cazenavette et al., 2024)	99.88	99.83	99.88	99.72
z_0	99.38	99.30	99.45	99.33
INC	**99.92**	**99.90**	**99.92**	**99.80**

K.5 Qualitative Error Analysis

Fig. 8 depicts examples of erroneous classifications made by DIRE and INC. We display examples of both false positive and false negative classifications. No consistent visual pattern distinguishes misclassified images from correctly classified ones, suggesting that classification errors are not attributable to obvious image artifacts.

L Limitations

Despite providing a theoretical and statistical account of why $\boldsymbol{z}_T$ benefits AIGI detection, our work has several limitations. First, the theoretical analysis is derived for deterministic DDIM inversion in

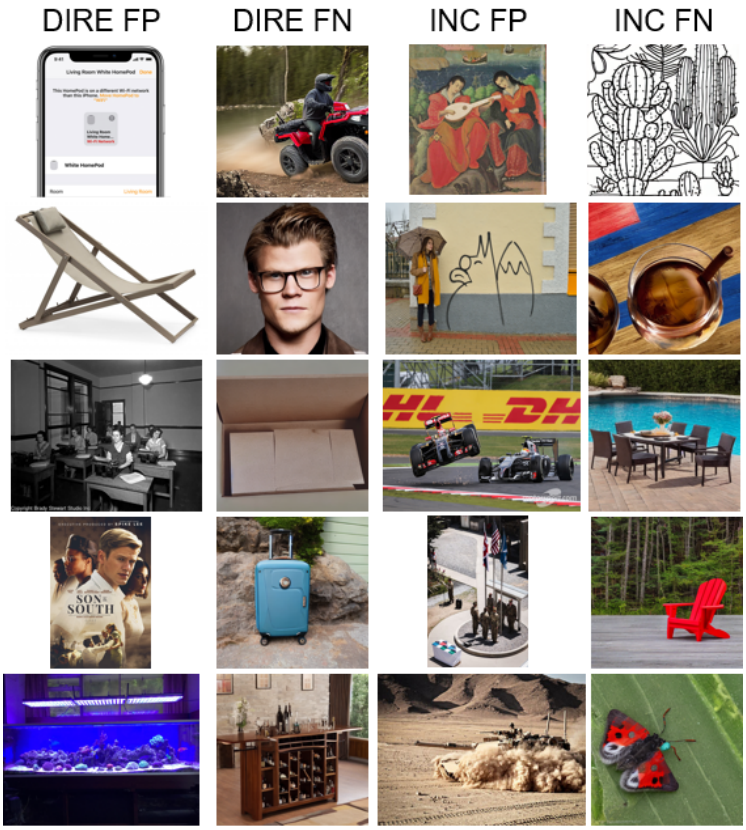


Figure 8: Examples of false positive (FP) and false negative (FN) classifications by DIRE and INC. False positives are real images incorrectly classified as generated, and false negatives vice versa.

the noise-prediction setting, which matches the formulation used in this work. Although the same intuition may extend to other inversion schemes, schedulers, or parameterizations, these cases are not studied explicitly. Second, several analyses rely on patchwise approximations and low-order statistics, including energy, mean, covariance, skewness, and KL divergence, rather than a full characterization of the high-dimensional latent distribution, since reliable density estimation is difficult with limited data. Third, our empirical study mainly focuses on latent diffusion and flow models with Gaussian terminal priors, so extensions to other generative families remain an open direction. Finally, although LAION-10k provides a diverse evaluation setting, our experiments are still limited in dataset coverage, and broader validation on additional datasets is left for future work.

M BROADER IMPACTS

On the positive side, the method can support misinformation mitigation, provenance analysis, and forensic verification of AIGIs. In particular, our method is more robust across diffusion models and image distortions, which may help practitioners identify AIGIs in more realistic deployment settings. It is also more interpretable. Understanding the difference between real and generated images in the noise domain may provide useful insights for image editing and related applications. On the negative side, the paper notes risks from decisions based on false positives or false negatives, and adversarial adaptation (e.g., using the released analysis to design more evasive generation pipelines). We emphasize that the detector should be used as probabilistic evidence rather than a definitive decision tool.

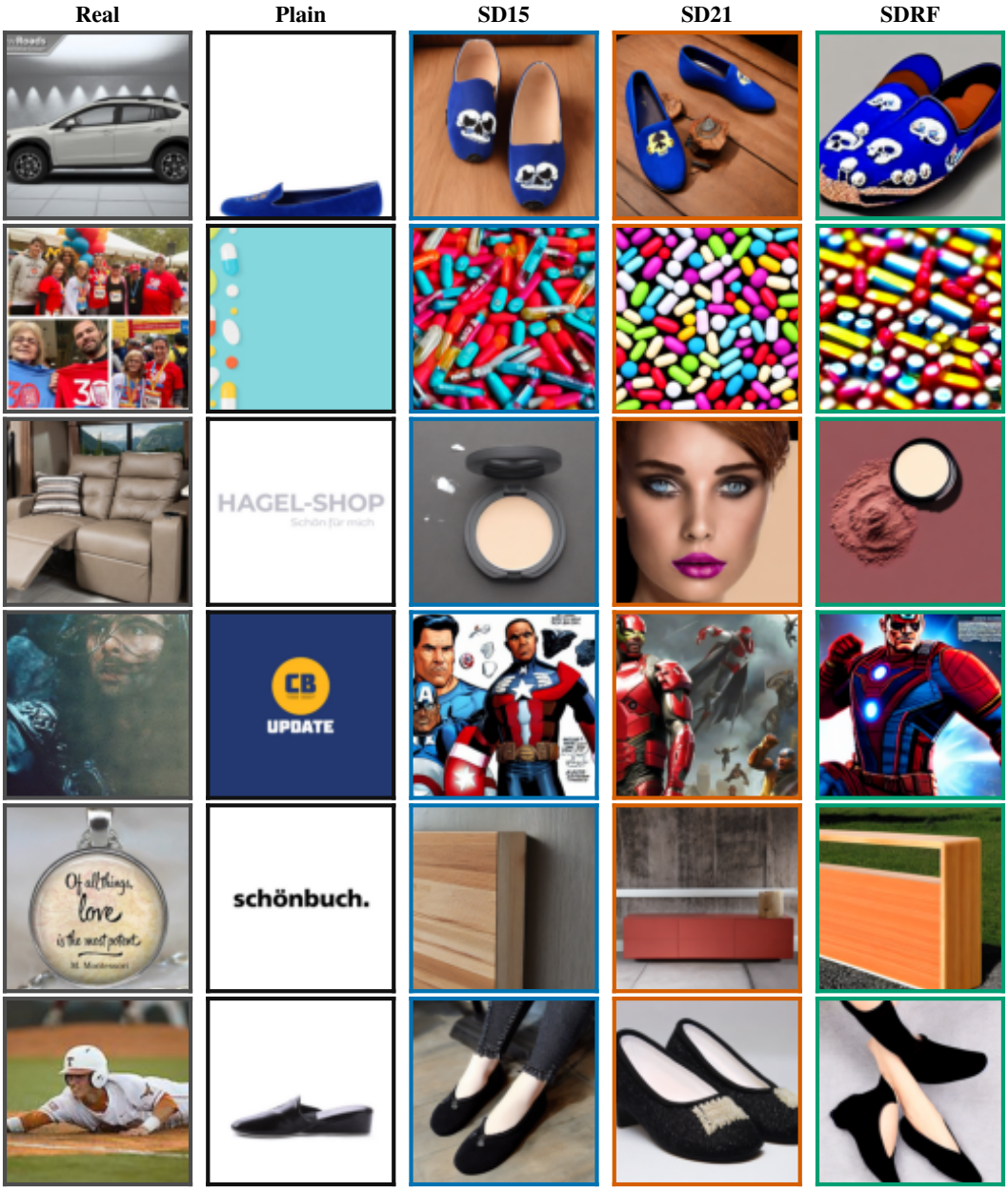


Figure 9: Images samples of real, real w/ plain, and generated images. The first column shows real images without plain pixels. The second column shows real images with more than 10% plain pixels; their inverted terminal latents z_T contain many near-zero-energy patches and deviate strongly from the Gaussian-prior expectation. The third to fifth columns show the corresponding generated versions of these plain-pixel images from SD1.5, SD2.1, and SDRF.