Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Unmasking Weaknesses: Adversarial Masks for Latent Diffusion Inpainting

Vlad Vasilescu

vlad.vasilescu2111@upb.ro

SIGMA Lab, CAMPUS Research Institute, National University of Science and Technology POLITEHNICA Bucharest https://orcid.org/0000-0003-3743-4235

Ana Neacșu

SIGMA Lab, CAMPUS Research Institute, National University of Science and Technology POLITEHNICA Bucharest

Jean-Christophe Pesquet

Centre de Vision Numerique, Inria, Universite Paris-Saclay, CentraleSupelec

Research Article

Keywords: Adversarial Attacks, Inpainting, Latent Diffusion Models, Adversarial Robustness

Posted Date: May 26th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-9806214/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: The authors declare no competing interests.

Unmasking Weaknesses: Adversarial Masks for Latent Diffusion Inpainting

Vlad Vasilescu[1*], Ana Neacșu[1] and Jean-Christophe Pesquet[2]

[1*]SIGMA Lab, CAMPUS Research Institute, National University of Science and Technology POLITEHNICA Bucharest, Bucharest, Romania.
[2]Centre de Vision Numérique, Inria, Université Paris-Saclay, CentraleSupélec, Gif-sur-Yvette, France.

*Corresponding author(s). E-mail(s): vlad.vasilescu2111@upb.ro;
Contributing authors: ana_antonia.neacsu@upb.ro;
jean-christophe.pesquet@centralesupelec.fr;

Abstract

Generative models have profoundly reshaped the creative and technical landscape by enabling the synthesis of high-fidelity content across a wide range of domains. However, theoretical studies on their stability are scarce, despite the critical importance of this topic for reliable real-world deployment. In this work, we address this challenge in the context of image inpainting using latent diffusion models. We first introduce a novel $\boldsymbol{\ell_0}$ adversarial attack that targets the inpainting mask and effectively addresses the associated binary optimization problem. We then conduct a detailed sensitivity analysis of the backward diffusion process to identify the most sensitive timestep in DDPM-based models. Finally, building on these insights, we propose several strategies to enhance robustness against adversarial attacks on diffusion models, including a new step-wise sensitivity regularization objective. Experiments on multiple benchmark datasets and well-established latent diffusion models demonstrate both the effectiveness of the proposed attack and the significant stability gains achieved with our defense mechanisms. Code will be released upon acceptance.

Keywords: Adversarial Attacks, Inpainting, Latent Diffusion Models, Adversarial Robustness

1 Introduction

Generative diffusion models (Ho et al. 2020; Song et al. 2021) have facilitated multiple applications in vision (Khader et al. 2023; Li et al. 2024b; Chen et al. 2024a), audio (Huang et al. 2023; Liu et al. 2023), or code generation (Singh et al. 2023; Gong et al. 2025), offering high-quality content at seemingly low cost. However, relatively little attention has been given to the stability of such models against input perturbations, which remains an open problem even for significantly simple neural networks (Bastounis et al. 2021). Following the emergence of Latent Diffusion Models (LDMs) / Stable Diffusion (Rombach et al. 2022), several works have addressed *adversarial attacks* against text-to-image models (Liang et al. 2023; Ma et al. 2025; Cheng et al. 2025), highlighting their susceptibility to word-level noise (Du et al. 2023). For image restoration, e.g., inpainting or super-resolution, the sensitivity of restored images with

respect to conditioning information remains relatively unexplored, with the sole prior work of Yu et al. (2024b) offering interesting empirical findings. In terms of robustness, previous works have addressed the problem through the lenses of noise schedules (Lin et al. 2024), latent space smoothness (Guo et al. 2024), or gradient smoothing (Wu et al. 2025). However, these works aim for overall quality improvement, rather than enforcing stability with respect to input noise. This stability issue is not yet fully explored in current research, highlighting an urgent need to investigate robustness more thoroughly.

Image inpainting (Suvorov et al. 2022; Corneanu et al. 2024) provides a particularly suitable and practically relevant testbed for studying stability under input perturbations. Unlike unconditional generation, inpainting is explicitly conditional: the model must synthesize content that is simultaneously (i) semantically consistent with the prompt (when present) (Chen et al. 2024b), and (ii) geometrically consistent with the unmasked context (Zhao et al. 2023). Such conditioning makes inpainting inherently sensitive to mask changes, resulting in significantly different reconstructions even for nearly-identical masks.

In this work, we address the sensitivity of latent inpainting diffusion models and develop a new adversarial attack targeting the inpainting mask, intended to alter the masked region and optimally disrupt the inpainting process. We investigate a previously overlooked failure mode of LDMs: small, structured perturbations to the binary inpainting mask can cause severe degradation in the reconstructed output. Since such imperfections arise naturally in real pipelines, through annotation noise, segmentation errors, or pre-processing artifacts, quantifying this sensitivity is a necessary step toward building more robust inpainting systems. We summarize our contributions as follows:

- We develop `MaskAttack`, an ℓ_0-constrained white-box adversarial attack for generating adversarial masks against inpainting LDMs (Section 3). We propose a simple yet effective heuristic solution to the binary optimization applied to the mask. Through extensive experiments on several diffusion models, we demonstrate that our attack successfully induces significant degradation in the generated visual features, achieving a higher adversarial effect than previous well-established attacks.
- We design a method for identifying *the most sensitive timestep* within DDPM backward processes (Subsection 3.5), fully leveraging the LDM susceptibility to noise prediction errors during adversarial mask optimization.
- Building on our findings, we propose several defense techniques aimed at training more robust LDMs, most notably a novel *stepwise sensitivity regularization* method for enhancing the robustness of LDMs against input perturbations (Subsection 4.2), increasing their resiliency in a versatile manner.

The rest of the paper is structured as follows. Section 2 covers previous works related to our study. Section 3 presents our attack algorithm along with the time-step sensitivity analysis. Section 4 discusses possible defenses and describes our proposed regularization strategy. Section 5 details our experimental setup, from both an attacker and a defender perspective. Finally, Section 6 concludes our work and discusses future developments.

2 Related Work

Adversarial Attacks. Prior work demonstrates that diffusion-based generative models are susceptible to carefully crafted adversarial examples that interfere with the backward process, effectively preventing models from learning or reproducing protected visual content (Liang et al. 2023; Dai et al. 2023; Zhang et al. 2025). Complementary studies reveal that the latent representations of generative systems inherently encode unsafe semantic directions (Schramowski et al. 2023; Vice et al. 2025), which can be exploited through structured prompt manipulation to bypass safety constraints and induce harmful or undesired outputs (Ma et al. 2025). This phenomenon aligns with broader observations that large generative models remain vulnerable to prompt-based attacks despite alignment efforts (Wei et al. 2023; Zou et al. 2023). Another very particular line of research aims to leverage diffusion models to purify input images in order to achieve adversarial robustness (Nie et al. 2022; Li et al. 2024a). However, even such systems have been demonstrated to be easily bypassed by evasion attacks

(Kang et al. 2023). Multimodal adversarial strategies further show that combining textual and visual perturbations can significantly amplify the effectiveness of jailbreak attacks, even under limited access assumptions (Qi et al. 2024; Ma et al. 2025). Extending this perspective to task-oriented settings, Yu et al. (2024b) introduced the *Mean Fluctuation Attack* (MFA), acting against conditional image restoration models and showing that intermediate denoising steps exhibit varying sensitivity.

Immunization Methods. Building upon the capabilities of adversarial examples, a significant body of literature has addressed the issue of immunizing the input data to prevent unauthorized or malicious editing via generative models. Salman et al. (2023) developed one of the precursor methods on immunization against diffusion-based image editing, optimizing imperceptible adversarial noise to disrupt text-based editing of a specific image. Building on this direction, Son et al. (2024) proposed a more efficient framework in which only specific timesteps within the backward diffusion are employed during optimization, improving convergence to stronger protective perturbations and reducing computational cost. Additionally, Choi et al. (2025) observed that targeting the early stages of the denoising process to optimize protective perturbations resulted in more efficient immunization strategies. AdvPaint (Jeon et al. 2025) similarly attends to inner self- and cross-attention blocks' activations, thus attending to both semantic representations and prompt information. Recently, Ozden et al. (2025) have proposed an optimization-free immunization framework, in which a separate immunization module is trained in advance and subsequently used as a pre-processing step before image editing. It is worth noting that, although both immunization methods and adversarial attacks involve crafted perturbations, their goals are distinct: immunizers seek to disrupt malicious generation, while our adversarial perturbations are designed to analyze and expose structural fragility in the generative reconstruction process.

Stability of generative models. Recent works on generative models have highlighted a shift from solely improving generation quality towards examining their instability (Du et al. 2023; Fotiadis et al. 2023). One research direction addressed the mismatch between training and inference by redesigning the sampling process to improve data generation stability and fidelity, either through novel noise schedules (Lin et al. 2024) or by leveraging additional internal information (Hong et al. 2023). Another challenge is latent space smoothness, previously tackled by regularizers that reduce sensitivity and enhance interpolation capabilities (Guo et al. 2024). Object hallucinations represent another pitfall of currently powerful multimodal models (Liu et al. 2024), with mitigation techniques including visual enhancement during modality fusion (Yin et al. 2025; Cao et al. 2025) and early detection (Betti et al. 2026). More recent methods explore the coordination between prior and likelihood updates during sampling, applying gradient smoothing to enhance overall robustness (Wu et al. 2025). Despite this recent progress, the challenge of simultaneously achieving stability and adversarial robustness in generative systems remains largely unexplored.

3 Proposed Framework

3.1 Background

Inpainting. We consider an image $x \in \mathcal{X}$, where $\mathcal{X}$ is some signal space. In computer vision, usually $\mathcal{X} \in \mathbb{R}^{H \times W \times C}$, where H and W are the height and width, and C is the number of channels. Image inpainting aims to estimate the full image x from masked representation $y = (1-m)^{\otimes C} \odot x$, where $m \in \{0,1\}^{H \times W}$ is the inpainting mask, $(\cdot)^{\otimes C}$ indicates that m is applied to all the channels, and $\odot$ designates the Hadamard product.

DDPM. Diffusion models are a leading alternative to classic inpainting methods, treating the task as a probabilistic generative process (Lugmayr et al. 2022; Anciukevičius et al. 2023). Most notably, DDPMs (Ho et al. 2020) are characterized by two main components: the forward and backward processes. Considering a timestep $t \in \{1, \dots T\}$ and $(\beta_1, \cdots, \beta_T) \in]0,1[^T$ the variance schedule across steps, the forward process constructs noisy observations from clean samples $x_0 \in \mathcal{X}$ as follows:

$$x_t = \sqrt{\bar{\alpha}_t} x_0 + \sqrt{1-\bar{\alpha}_t}\eta_t, \tag{1}$$

where $\eta_t \sim \mathcal{N}(0, \mathrm{I})$ and $\bar{\alpha}_t = \prod_{i=1}^{t}(1-\beta_i)$. The backward process estimates the added noise using

a neural network $\epsilon_\theta : \mathcal{X} \times \{0, \dots, T-1\} \to \mathcal{X}$ parametrized by θ:

$$x_{t-1} = \frac{1}{\sqrt{1-\beta_t}}\Big(x_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\widehat{\eta}_t\Big) + \sqrt{\beta_t}\zeta_t, \quad (2)$$

where $\zeta_t \sim \mathcal{N}(0, \mathrm{I})$ and $\widehat{\eta}_t = \epsilon_\theta(x_t, t)$, training being performed using standard MSE. Subsequent works developed efficient sampling methods, significantly reducing the number of function evaluations (Song et al. 2020; Lu et al. 2022).

Latent Diffusion Models (LDMs). Rombach et al. (2022) proposed to enhance the scalability of generating high-resolution content by moving the denoising process to a latent space spanned by an encoder-decoder structure $\mathcal{D} \circ \mathcal{E}$. More precisely, a clean sample $x_0 \in \mathcal{X}$ is encoded as $z_0 = \mathcal{E}(x_0) \in \mathbb{R}^{\frac{H}{s} \times \frac{W}{s} \times C'}$ for a downsampling factor $s > 1$, with noisy observations z_t computed similarly to Eq. (2). The reconstructed latent sample $\widehat{z}_0$ is then mapped back to the image space by $\widehat{x}_0 = \mathcal{D}(\widehat{z}_0) \in \mathbb{R}^{H \times W \times C}$.

For inpainting, the denoising operator ϵ_θ is additionally conditioned on the unmasked pixels $(1-m)^{\otimes C} \odot x_0$. Following the same protocol as Suvorov et al. (2022), inpainting LDMs use a denoising network $\epsilon_\theta : \mathcal{X} \times \{0, \dots, T-1\} \times \mathcal{X} \to \mathcal{X}$, with an additional input z_c given by

$$z_c(x, m) = \begin{bmatrix} m \downarrow_s \\ \mathcal{E}\big((1-m)^{\otimes C} \odot x\big) \end{bmatrix}, \quad (3)$$

with $z_c(x, m) \in \mathbb{R}^{\frac{H}{s} \times \frac{W}{s} \times (C'+1)}$, and $(\cdot) \downarrow_s$ designates downsampling with factor s. In this case, the backward process from Eq. (2) becomes

$$\widehat{\eta}_t = \epsilon_\theta\big(z_t, t, z_c(x, m)\big). \quad (4)$$

This pushes the network to account for the available known information encompassed in $z_c(x, m)$.

3.2 Proposed attack algorithm

We focus on *white-box untargeted* adversarial attacks that craft minor perturbations applicable to the original inpainting mask, denoted by $\overline{m}$. This way, we seek to demonstrate the full extent of a model's vulnerability under the strongest threat model. While many existing attacks focus on perturbing the input image, in real-world contexts, the mask itself is even more susceptible to perturbations, as it can be affected by factors such as user inaccuracies during manual annotations, errors introduced by automated segmentation algorithms, or other artifacts that cause variations in the mask. Specifically, we are interested in finding $m_{\text{adv}} \in \{0,1\}^{H \times W}$ which maximizes the adversarial objective $\mathcal{L}_{\text{attack}}$, under budget $\delta > 0$:

$$m_{\text{adv}} = \underset{\substack{m \in \{0,1\}^{H \times W} \\ \|m - \overline{m}\|_0 \leq \delta}}{\arg\max} \mathcal{L}_{\text{attack}}(\widetilde{\eta}_t, \eta_t), \quad (5)$$

where $\|\cdot\|_0$ is the ℓ_0 pseudo-norm and

$$\widetilde{\eta}_t = \epsilon_\theta\big(z_t, t, \widetilde{z}_c(x, m)\big), \quad (6)$$

$$\widetilde{z}_c(x, m) = z_c(x, \mathcal{F}(m)). \quad (7)$$

Hereabove, $\mathcal{F}$ is a (possibly nonlinear) spatial filter which is applied to the mask. The role of $\mathcal{F}$ will be discussed in Subsection 3.4. The ℓ_0 pseudo-norm constraint in Eq. (5) corresponds to the maximum number of allowed binary changes in $\overline{m}$. Unlike MFA, which allows for a maximum modification δ at each pixel in x, our attack is more restrictive and searches for a limited number of pixels that can be modified. We include a detailed discussion regarding MFA's algorithm in Appendix A for reference.

Algorithm 1 `MaskAttack`, ℓ_0 norm

Input: Clean image x, mask $\overline{m}$, attack steps N, budget δ, objective $\mathcal{L}_{\text{attack}}$, number of trials K, filtering operator $\mathcal{F}$, t_{attack} (optional)

Output: Adversarial mask $m_{\text{adv}}^{(N)}$

$z_0 = \mathcal{E}(x)$, $m_{\text{adv}}^{(0)} = \overline{m}$
Partition budget $\delta = \sum_{n=1}^{N} \delta_n$, with $\delta_n > 0$
for $n = 0, \dots, N-1$ **do**
  $\widetilde{z}_c^{(n)} = z_c\big(x, \mathcal{F}(m_{\text{adv}}^{(n)})\big)$
  $t_n \sim \mathcal{U}(0, T)$ or $t_n = t_{\text{attack}}$
  $g^{(n)} = 0$ `// gradient accumulator`
  for $k = 1, \dots, K$ **do**
    $\eta^{(n,k)} \sim \mathcal{N}(0, \mathrm{I})$
    $z^{(n,k)} = \sqrt{\bar{\alpha}_{t_n}} z_0 + \sqrt{1-\bar{\alpha}_{t_n}} \eta^{(n,k)}$
    $\widetilde{\eta}^{(n,k)} = \epsilon_\theta(z^{(n,k)}, t_n, \widetilde{z}_c^{(n)})$
    $g^{(n)} \leftarrow g^{(n)} + \nabla_m \mathcal{L}_{\text{attack}}(\widetilde{\eta}^{(n,k)}, \eta^{(n,k)})$
  end for
  $g^{(n)} \leftarrow g^{(n)} / K$ `// statistical averaging`
  $m_{\text{adv}}^{(n+1)} =$ `UpdateMask(`$m_{\text{adv}}^{(n)}$`,` $g^{(n)}$`,` δ_n`)`
end for

Algorithm 2 `UpdateMask`

Input: Current mask m_{adv}, gradient g, budget δ_n
Output: Updated adversarial mask m'_{adv}

$m_{\text{s}} = \text{proj}_{\{0,1\}^{H\times W}}\left(m_{\text{adv}} + \text{sgn}(g)\right)$
$d = m_{\text{adv}} \oplus m_{\text{s}}$ // XOR detection of changed values
$g_{\text{m}} = \text{abs}(g) \odot d$ // possible candidates
$i = \text{top}(g_m, \delta_n)$ // δ_n indices with highest gradients
$m'_{\text{adv}} = m_{\text{adv}}$
$m'_{\text{adv}}[\,i\,] \leftarrow 1 - m'_{\text{adv}}[\,i\,]$

Algorithm 1 encompasses our `MaskAttack`, providing a simple heuristic solution to the binary optimization problem in Eq. (5). Such problems have previously been addressed by methods requiring specific assumptions on the objective (Kolmogorov and Rother 2007), or in some cases result in slow and unscalable processes (Büsing et al. 2023). Our gradient-guided flipping strategy within an ℓ_0 budget constraint is specifically designed for this combinatorial structure, and its effectiveness will be empirically validated within the experimental section.

Despite sharing a basic framework with `MFA`, our approach introduces several key aspects that make it very efficient. First, we partition the total budget δ into per-attack-step budgets $(\delta_n)_{1\leq n\leq N}$, allowing for a finer craft of the final adversarial mask m_{adv}. Second, we devise an `UpdateMask` mechanism (shown in Algorithm 2) which performs changes on the highest-magnitude gradients *only if* they incur changes in the current mask. In this way, we avoid standard first-order strategies which would add real-valued gradients $\nabla_m \mathcal{L}_{\text{attack}}$ directly to m_{adv}, so requiring an additional binarization step afterwards. This prevents chaotic updates on $m^{(n)}_{\text{adv}}$ or the algorithm getting stalled due to small gradient values. More precisely, at each attack step n, based on the sign and magnitude of $\nabla_m \mathcal{L}_{\text{attack}}$, the update procedure changes at most δ_n pixels from the current mask, among those which i) could result in a change of mask value, and ii) yield the highest gradient magnitude.

3.3 Attack Objective

We consider two choices for $\mathcal{L}_{\text{attack}}$:

$$\texttt{MFA}: \widetilde{\mathcal{L}}_{\text{attack}}(\widetilde{\eta}_t, \eta_t) = \mathsf{average}\big[\widetilde{\eta}_t\big], \tag{8}$$

$$\texttt{EM}: \widetilde{\mathcal{L}}_{\text{attack}}(\widetilde{\eta}_t, \eta_t) = \|\eta_t - \widetilde{\eta}_t\|_2^2. \tag{9}$$

The Error-Maximization (`EM`) loss is equivalent to the one used by AdvDM (Liang et al. 2023). In our case, we found that forcing the objective function to focus only on the inpainted region yields better results. In addition, to allow the attack to be successful for a wide range of images, we included an expectation with respect to the probability distribution of x. This translates into the final form of our objective:

$$\mathcal{L}_{\text{attack}}(\widetilde{\eta}_t, \eta_t) \triangleq \mathbb{E}_x[\widetilde{\mathcal{L}}_{\text{attack}}(\widetilde{\eta}_t \odot \widetilde{m},\, \eta_t \odot \widetilde{m})], \tag{10}$$

where $\widetilde{m} = d(\overline{m}\downarrow_s)^{\otimes C}$ and $d(\cdot)$ denotes the morphological dilation operator with a small structuring element. The Hadamard product with the dilated version of $\overline{m}$ will allow to update m_{adv} only within the slightly-enlarged masked region, which will be the focus of the attack. The gradient of $\mathcal{L}_{\text{attack}}$ is given by the chain rule as follows:

$$\begin{aligned}\nabla_m \mathcal{L}^{(n)}_{\text{attack}} &= \mathbb{E}_x[\nabla_m \widetilde{\mathcal{L}}^{(n)}_{\text{attack}}] \\ &= \mathbf{J}^{\widetilde{\mathcal{L}}_{\text{attack}}}_{\widetilde{\eta}_t}(\widetilde{\eta}^{(n)} \odot \widetilde{m}, \eta^{(n)} \odot \widetilde{m}) \\ &\times \mathbf{J}^{\widetilde{\eta}_t}_{\widetilde{z}_c}(z^{(n)}, t_n, \widetilde{z}^{(n)}_c)\, \mathbb{E}_x[\mathbf{J}^{\widetilde{z}_c}_m(x, \mathcal{F}(m^{(n)}_{\text{adv}}))]\, \mathbf{D}_{\widetilde{m}},\end{aligned} \tag{11}$$

where $\mathbf{D}_m = \text{Diag}[(1-m)^{\otimes C}]$. A further simplification arises from the assumption that

$$\mathbb{E}_x[\mathbf{J}^{\mathcal{E}}\big((1 - \mathcal{F}(m^{(n)}_{\text{adv}})) \odot x\big)\mathbf{D}_x] = \mathbf{0}. \tag{12}$$

See Appendix B for a justification. In particular, this assumption is satisfied if, for every m, the probability distribution of $\mathbf{J}^{\mathcal{E}}\big((1-\mathcal{F}(m)) \odot x\big)\mathbf{D}_x$ is centered and symmetric. This assumption yields

$$\mathbb{E}_x[\mathbf{J}^{\widetilde{z}_c}_m(x, \mathcal{F}(m^{(n)}_{\text{adv}}))]\mathbf{D}_{\widetilde{m}} = \begin{bmatrix} \mathbf{J}^{(\cdot)\downarrow_s}\, \mathbf{J}^{\mathcal{F}}(m^{(n)}_{\text{adv}})\, \mathbf{D}_{\widetilde{m}} \\ \mathbf{0} \end{bmatrix}. \tag{13}$$

Due to the specific form of the subsampling process, $\mathbf{J}^{(\cdot)\downarrow_s}$ is actually a sparse binary matrix in the above expression. Plugging Eq. (13) in Eq. (11) shows that it is only necessary to compute the

Jacobian of $\widetilde{\eta}_t$ with respect to the first $H/s \times W/s$ components of $\widetilde{z}_c$, thus significantly reducing the computational cost. Another benefit of the proposed formulation is that it avoids differentiating $\mathcal{E}$. Indeed, in the recent work of Ramanaik et al. (2025), it was shown that ill-conditioning may be present within encoder-decoder structures, hindering the attack's optimization path.

3.4 Choice of filtering operator $\mathcal{F}$

If no filtering is performed ($\mathcal{F} = \mathsf{Id}$), due to the subsampling conducted in Eq. (3), the attacker tends to modify only isolated pixels, resulting in an adversarial mask that may appear unrealistic (as it will be shown in Section 5). A simple way to avoid this effect is to introduce some spatial smoothing through $\mathcal{F}$ to steer the optimization towards non-isolated mask changes. For $\mathcal{F}$, we adopt the *Soft-Median Function* (SMF) (Zhao et al. 2020) as a differentiable surrogate for the median filter.

Remark 1 An alternative solution to enforce the generation of more homogeneous areas in the adversarial mask consists of incorporating a smoothed isotropic Total Variation (TV) term in the objective function Eq. (10), of the form:

$$\mathcal{L}_{\text{TV}}(m) = -\lambda_{\text{TV}} \sum_{w=1}^{W-1} \sum_{h=1}^{H-1} \Delta_{h,w}(m), \tag{14}$$

where $\lambda_{\text{TV}} > 0$, $\Delta_{h,w}$ is a smoothed version of the norm of the spatial gradient, defined as

$$\Delta_{h,w}(m) = \sqrt{A_{h,w} + B_{h,w} + \eta^2}, \tag{15}$$

where $A_{h,w} = (m_{h,w+1} - m_{h,w})^2$, $B_{h,w} = (m_{h+1,w} - m_{h,w})^2$, and $\eta > 0$ is the smoothing parameter. According to our experiments from Section 5, this regularization approach, however, turned out to be less efficient than introducing SMF.

3.5 Optimal attack timestep

Given that $\mathcal{L}_{\text{attack}}$ acts at each iteration on a single-timestep prediction $\widetilde{\eta}_t$, while the end-goal of an attacker is to maximally perturb the fully-reconstructed z_0, running an efficient attack heavily relies on choosing the timestep $t^* \in \{1, \dots, T\}$ where $\widetilde{\eta}_{t^*}$ has the maximum influence over z_0. This timestep can be formalized as

$$t^* = \underset{t=\{1,\dots,T\}}{\arg\max} \, \|\mathbf{J}_{\widetilde{\eta}_t}^{z_0}\|_{\text{S}}. \tag{16}$$

In order to streamline the search, we expand the inner Jacobian using the chain rule, as follows:

$$\mathbf{J}_{\widetilde{\eta}_t}^{z_0} = \mathbf{J}_{z_1}^{z_0} \mathbf{J}_{z_2}^{z_1} \cdots \mathbf{J}_{z_{t-1}}^{z_{t-2}} \mathbf{J}_{\widetilde{\eta}_t}^{z_{t-1}}, \tag{17}$$

where, given Eq. (2), for every $k \in \{1, \dots, t\}$,

$$\mathbf{J}_{z_k}^{z_{k-1}} = \frac{1}{\sqrt{1-\beta_k}} \big(\mathbf{I} - \frac{\beta_k}{\sqrt{1-\bar{\alpha}_k}} \mathbf{J}_{z_k}^{\widetilde{\eta}_k}\big), \tag{18}$$

$$\mathbf{J}_{\widetilde{\eta}_k}^{z_{k-1}} = -\frac{\beta_k}{\sqrt{(1-\beta_k)(1-\bar{\alpha}_k)}} \mathbf{I}. \tag{19}$$

We can upper bound the norm of $\mathbf{J}_{\widetilde{\eta}_t}^{z_0}$ as follows:

$$\|\mathbf{J}_{\widetilde{\eta}_t}^{z_0}\|_{\text{S}} = \Big\| \big(\prod_{k=1}^{t-1} \mathbf{J}_{z_k}^{z_{k-1}} \big) \mathbf{J}_{\widetilde{\eta}_t}^{z_{t-1}} \Big\|_{\text{S}} \tag{20}$$

$$\leq \Big(\prod_{k=1}^{t-1} \|\mathbf{J}_{z_k}^{z_{k-1}}\|_{\text{S}} \Big) \|\mathbf{J}_{\widetilde{\eta}_t}^{z_{t-1}}\|_{\text{S}} \triangleq s(t). \tag{21}$$

We define $s(t)$ as the *timestep sensitivity* variable, which, after substituting the Jacobians by their expressions in Eqs. (18) and (19), is expressed as

$$s(t) = \frac{\beta_t}{\sqrt{\bar{\alpha}_t(1-\bar{\alpha}_t)}} \prod_{k=1}^{t-1} \Big\| \mathbf{I} - \frac{\beta_k}{\sqrt{1-\bar{\alpha}_k}} \mathbf{J}_{z_k}^{\widetilde{\eta}_k} \Big\|_{\text{S}}. \tag{22}$$

Finally, we can estimate the timestep whose predicted noise has the strongest influence over z_0 by maximizing the upper bound in Eq. (21):

$$t^* \approx \underset{t=\{1,\dots,T\}}{\arg\max} \, s(t). \tag{23}$$

The sensitivity variable from Eq. (22) can be recursively computed as

$$s(t+1) = s(t) \frac{\gamma_{t+1}}{\gamma_t} \Big\| \mathbf{I} - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}} \mathbf{J}_{z_t}^{\widetilde{\eta}_t} \Big\|_{\text{S}}, \tag{24}$$

with $\gamma_t = \frac{\beta_t}{\sqrt{\bar{\alpha}_t(1-\bar{\alpha}_t)}}$. This recursive computation eliminates the need to directly differentiate z_0. In Section 5, we provide an empirical analysis regarding per-timestep sensitivities (see Figure 1), indicating a specific time interval where the backward process is more sensitive w.r.t. ϵ_θ's predictions. Our results align with previous shallow empirical findings of Chen et al. (2023); Huang et al. (2024), in which it was generally accepted

that early denoising steps have a greater influence over the final performance. For an alternative analysis regarding the sensitivity of the backward process with respect to the mask m, we refer the reader to Appendix C.

Remark 2 The bound from Eq. (21) is tight when the singular vectors of consecutive Jacobians $\mathbf{J}_{z_k}^{z_{k-1}}$ are co-aligned across timesteps. Intuitively, this condition can also be reflected in noise predictions $\widetilde{\eta}_k$ varying smoothly along the backward trajectory, which is usually the case in DDPM models utilizing a sufficiently larger number of timesteps T. Empirically, in Section 5, we will prove the bound's practical utility: using $t_{\text{attack}} = t^*$ in Algorithm 1 yields the highest degradation, while neighboring timesteps lead to comparable attack performance, indicating a smooth behavior around our estimate (please refer to Table 5).

Remark 3 Through `MaskAttack`, we aim to highlight a qualitatively different threat surface rather than claim budget-matched superiority. Since ℓ_0 and ℓ_∞ norms represent fundamentally distinct perturbation types, they cannot be mapped to a common scale. This being said, the choice of our ℓ_0 budgets in the experimental section is conservative and practically motivated. On the other hand, for pixel-based attacks, we follow the budget values reported in their respective papers, reflecting each method's performance as originally proposed.

4 Proposed Defenses

In Section 5, it will be shown that standardly-trained inpainting LDMs are highly susceptible to the previously devised attack, and even to random mask modifications. Hence, in this section, we discuss several possible improvements to the training algorithm of standard LDMs, aimed at increasing the resilience of the generative process.

4.1 Naïve Defenses

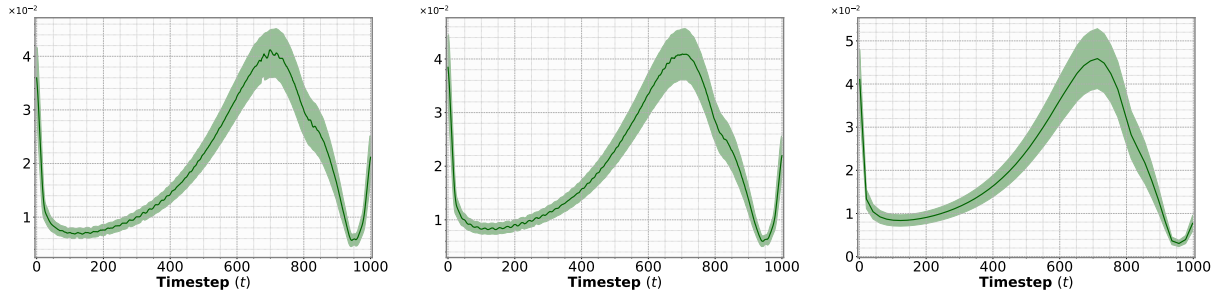
Adversarial Training (AT) has been considered the go-to defense in many security-related applications (Shafahi et al. 2019). However, including a multi-step gradient-based attack when training diffusion models would be highly impractical. In the recent work of Rosaria et al. (2025), AT was extended for unconditional diffusion models; nonetheless, this resulted in a significant performance drop on clean data, even for simple benchmarks. Moreover, in inpainting pipelines, masks are usually generated on-the-fly at each training step to increase generalization. This naturally rules out the use of AT, given the ever-changing nature of training data. Hence, as a baseline defense method, we simply consider training with randomized masks m – that is, with probability $p \in (0, 1]$ we modify δ random pixels in the original mask m. We symbolize this *Randomized-mask Training* with $\texttt{RT}_p^\delta$.

Algorithm 3 Forward training step of `TIR`

Input: Image-mask pair (x, m), mask transform operator $\mathcal{T}_p$, consistency factor $\tau \in \mathbb{R}_+$
Output: Total training loss $\mathcal{L}$

$t \sim \mathcal{U}(1, T)$, $\eta_t \sim \mathcal{N}(0, I)$
$z_t = \sqrt{\bar{\alpha}_t}\mathcal{E}(x) + \sqrt{1-\bar{\alpha}_t}\eta_t$
$z_c^m = z_c(x, m)$, $z_c^{m_p} = z_c\big(x, \mathcal{T}_p(m)\big)$
$\widehat{\eta}_{t,m} = \epsilon_\theta(z_t, t, z_c^m)$, $\widehat{\eta}_{t,m_p} = \epsilon_\theta(z_t, t, z_c^{m_p})$
$\mathcal{L}_c = \big\| \big(m \odot \mathcal{T}_p(m)\big) \odot \big(\widehat{\eta}_{t,m} - \widehat{\eta}_{t,m_p}\big) \big\|_2^2$
$\mathcal{L} = \|\eta_t - \widehat{\eta}_{t,m}\|_2^2 + \|\eta_t - \widehat{\eta}_{t,m_p}\|_2^2 + \tau\mathcal{L}_c$

While AT has been highly adopted, it has been shown to suffer from *robust overfitting* (Tang and Zhang 2024), i.e., learning the adversarial noise pattern. *Regularized Training* aims to mitigate the over-reliance on highly-specific noise patterns generated during training and enforce a more holistic idea of robustness, using additional loss terms to enforce certain characteristics. One simple approach would be to guide the network to generate similar outputs for a mask m and its transformed variant $m_p = \mathcal{T}_p(m)$, $p \in [0, 1]$, where $\mathcal{T}_p$ denotes a mask transform which preserves the mask's shape and applies shifts/rotations/shears such that an overlap of $(p \cdot 100)\%$ is kept between the original and transformed mask. We name this approach as *Transform Invariant Regularization* (TIR), illustrated by Algorithm 3. Given the same z_t, two forward passes are executed with two different conditions z_c. The goal is to align the network's responses when presented with visible contexts $x \odot (1-m)^{\otimes C}$ and $x \odot (1-\mathcal{T}_p(m))^{\otimes C}$, enforcing stable responses under natural mask changes.



(a) AFHQ-Cats (b) CelebA-HQ (c) LSUN-Churches

Fig. 1: Statistics $\big(\mu(t) \pm \sigma(t)\big)$ of per-timestep sensitivity $s(t)$ from (22) for standard LDMs trained on different datasets. Average statistics $\mu(t)$ and $\sigma(t)$ are computed over 1000 independent x_t (1), using randomly sampled x_0 from each dataset.

4.2 Sensitivity Regularization

Let us come back to the analysis performed to derive t^* from (23) and, more specifically, the recursive computation of the *timestep sensitivity* in (22). A valid objective is to enforce lower sensitivity near the initial generation steps ($t \to T$), which were shown to induce a significant effect on the final reconstruction (Chen et al. 2023; Huang et al. 2024; Yu et al. 2024b). We propose to translate this goal into the following condition:

$$(\forall t \in \{1, \dots, T-1\}) \quad s(t+1) \leq s(t), \tag{25}$$

leading to the inequality constraint:

$$\big\|\mathbf{I} - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}} \mathbf{J}_{z_t}^{\widetilde{\eta}_t}\big\|_{\mathrm{S}} \leq \frac{\gamma_t}{\gamma_{t+1}}. \tag{26}$$

Thus, in order to enforce (25), we define the following *Step-wise Sensitivity Regularization* (SSR) loss

$$\mathcal{L}_{\mathrm{SSR}}(z_t, t, \theta) \triangleq \lambda \Big[\big\|\mathbf{I} - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}} \mathbf{J}_{z_t}^{\widetilde{\eta}_t}\big\|_{\mathrm{S}} - \frac{\gamma_t}{\gamma_{t+1}}\Big]_+, \tag{27}$$

where $\lambda \geq 0$ denotes the regularization strength. As opposed to other Jacobian regularization schemes, which drive the spectral norm arbitrarily low, $\mathcal{L}_{\mathrm{SSR}}$ implies a softer penalty and leads to more stable regularized training when coupled with the standard denoising objective. Practically, at each training step, the same t randomly chosen to construct z_t is used within our regularization term from (27). Moreover, given that SSR is derived directly from the sensitivity analysis of the DDPM backward process, with its design grounded in the theoretical characterization of the most vulnerable timestep, this makes it a theoretically-motivated defense mechanism rather than an empirically tuned element.

5 Experiments

5.1 Setup

In our experiments, we use different variants of LDMs. We experimented on three highly used datasets, namely CelebA-HQ (Xia et al. 2021), LSUN-Churches (Yu et al. 2024a), and AFHQ-Cats (Choi et al. 2020), all at 256×256 resolution. Additionally, we include several experiments on extensively pre-trained LDM/Stable Diffusion models on the Places365 dataset (Zhou et al. 2017), at 512×512 resolution, to expand upon our findings. We compare our `MaskAttack` with two previously discussed image-based attacks, `MFA` (Yu et al. 2024b) and `AdvDM` (Liang et al. 2023). Details regarding architectures, training, inference, attacks, and defense configurations can be found in Appendix D.

To quantify the degradation induced by adversarial masks, we primarily rely on FID and LPIPS. FID measures the distributional shift between generated and reference images, capturing how far the degraded reconstructions deviate from realistic ones at a distributional level. LPIPS complements this by operating at the level of individual image pairs, measuring perceptual dissimilarity in a feature space that better reflects human visual

Table 1: Performance of LDM models trained under different settings. Bold indicates the best attack performance against each defense. Underscore denotes the most robust model against each attack.

Dataset	Defense	No attack		**MFA**		**AdvDM**		`MaskAttack-MFA`		`MaskAttack-EM`		$\texttt{Rand}_\delta$	
		FID ↓	LPIPS ↓	FID ↑	LPIPS ↑	FID ↑	LPIPS ↑	FID ↑	LPIPS ↑	FID ↑	LPIPS ↑	FID ↑	LPIPS ↑
CelebA-HQ	-	16.25	0.112	16.94	0.118	22.71	0.152	37.55	0.137	**61.95**	**0.154**	42.21	0.154
	$\texttt{RT}^{64}_{0.75}$	13.04	0.105	13.57	0.110	20.02	0.145	20.91	0.118	**62.09**	**0.149**	13.80	0.105
	`TIR`	15.52	0.113	15.74	0.118	19.23	0.141	30.24	0.135	**51.39**	**0.151**	38.93	0.151
	$\texttt{RT}^{64}_{0.75}$ + `TIR`	15.29	0.111	14.50	0.113	20.81	**0.154**	24.61	0.120	**50.93**	0.147	16.35	0.108
	$\mathcal{L}_{\text{SSR}}$	11.35	0.103	11.94	0.109	13.40	0.130	21.68	0.118	**37.06**	0.137	36.87	**0.143**
LSUN-Church.	-	11.11	0.161	13.24	0.173	14.58	0.186	15.41	0.205	**20.02**	0.215	17.76	**0.231**
	$\texttt{RT}^{64}_{0.75}$	11.25	0.160	12.83	0.173	13.65	0.179	13.64	0.189	**19.00**	**0.215**	11.56	0.159
	`TIR`	9.86	0.160	11.29	0.171	12.07	0.182	13.25	0.203	**18.30**	0.217	15.63	**0.229**
	$\texttt{RT}^{64}_{0.75}$ + `TIR`	9.27	0.157	10.31	0.169	11.09	0.170	11.24	0.185	**15.64**	**0.207**	8.85	0.147
	$\mathcal{L}_{\text{SSR}}$	9.96	0.156	11.55	0.166	11.63	0.170	11.94	0.185	14.76	0.202	**15.41**	**0.224**
AFHQ-Cats	-	12.71	0.133	13.28	0.143	14.83	0.166	16.27	0.161	**20.86**	**0.174**	19.85	0.167
	$\texttt{RT}^{64}_{0.75}$	13.17	0.134	14.13	0.143	13.63	0.152	14.54	0.144	**18.07**	**0.162**	12.99	0.134
	`TIR`	13.41	0.139	13.64	0.148	14.72	0.162	14.39	0.155	17.9	0.168	**19.84**	**0.168**
	$\texttt{RT}^{64}_{0.75}$ + `TIR`	12.81	0.127	12.77	0.137	**15.55**	**0.177**	12.82	0.128	14.96	0.143	12.13	0.124
	$\mathcal{L}_{\text{SSR}}$	10.93	0.120	12.47	0.135	13.76	**0.168**	11.47	0.127	16.38	0.154	**17.93**	0.155



Fig. 2: Attack examples from LSUN-Churches on an undefended LDM model. Top row illustrates the masked input images, and subsequent rows show reconstructions under different attacks.

judgment. These two metrics are particularly well-suited for the large-mask regime, where the inpainting task transitions from reconstruction to generation: in such cases, there is no single ground-truth completion, and reference-based pixel-level metrics lose their reliability. Nevertheless, PSNR and SSIM are reported in some cases as secondary metrics to provide a more comprehensive evaluation.

5.2 Attack performance

Table 1 illustrates the attack performance on LDMs optimized on CelebA-HQ, LSUN-Churches, and AFHQ-Cats. For comparison, we also compute the performance under δ random mask modifications, symbolized with $\texttt{Rand}_\delta$. It is clear that our `MaskAttack` achieves better performance, compared to its competitors, MFA and AdvDM, in all cases, highlighting its efficiency, while also

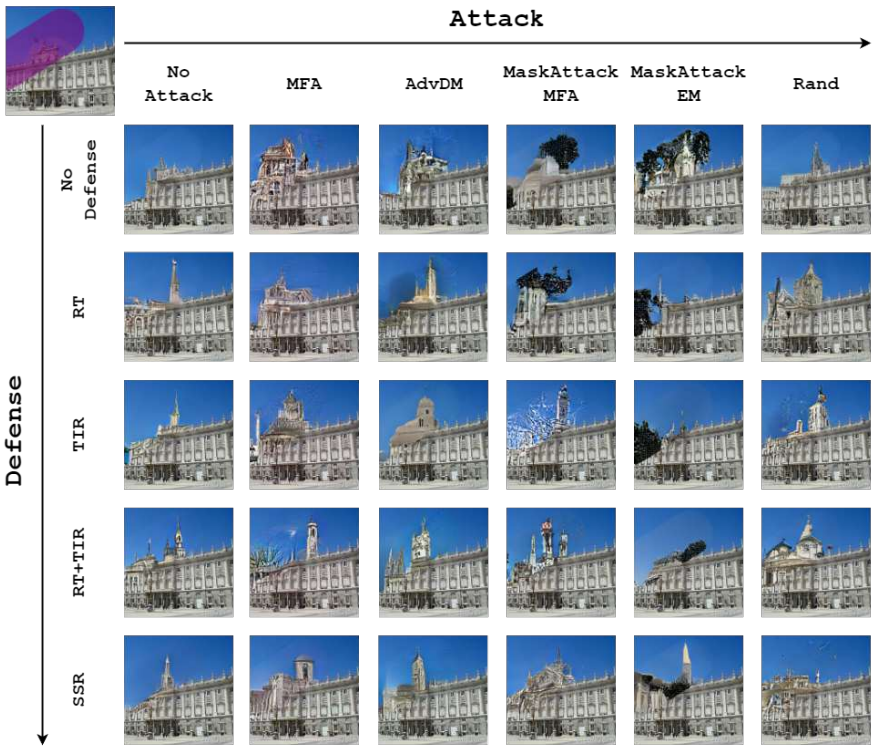


Fig. 3: Reconstruction comparison for an LSUN-Church example, under all pairs of attacks and defenses.

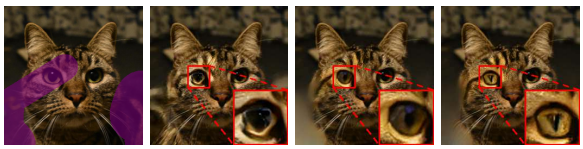
Table 2: Performance (FID↑) comparison for different encoder-decoder models and resolutions on CelebA-HQ.

Model	$\mathcal{E}-\mathcal{D}$	Clean	MFA	MaskAttack
LDM-4	VQ-VAE	24.07	34.38	**55.58**
	KL-VAE	16.25	16.94	**61.95**
LDM-8	VQ-VAE	20.01	37.65	**89.34**
	KL-VAE	20.37	23.02	**88.16**

indicating an LDM's higher susceptibility to mask modifications compared to pixel space attacks. Using the error maximization objective outperforms the mean fluctuation loss proposed by MFA, showcased by both our attack and AdvDM. This highlights that LDMs are actually more stable w.r.t mean predicted-noise fluctuations in the backward process than initially observed by Yu et al. (2024b), while maximizing EM, which is similar to the training objective, allows an attacker to easily revert a previously-learned trajectory and create more powerful attacks. Figure 2 shows several clean and adversarially-generated samples from LSUN-Churches, on the undefended LDM. Overall, MFA leads to a slight hue change in the reconstructed areas, while our attack inflicts significant degradation in the generated structure.

In Table 2 we compare the performance of several LDMs trained with different encoder-decoder structures, each attending to its own latent space representation. Using an increased downsampling factor s yields higher attack performance, indicating that while carefully-crafted pixel noise (MFA) has approximately a similar effect, simple pixel masking/unmasking induces significant changes in the latent space. This showcases a significant flaw of widely-used encoder-decoder structures, with higher compression rates leading to less stable latent representations.

In Figure 3 we showcase an example from LSUN-Churches, reconstructed under all possible configurations of attacks and defense models. While all defense techniques allow for good quality reconstruction when no attack is used, the undefended and `RT`-trained models lead to the highest variability between generated visual features during different attacks. Our `SSR` regularization



(a) Input (b) $\lambda = 0.01$ (c) $\lambda = 0.1$ (d) $\lambda = 1$

Fig. 4: Regularization effect on clean reconstruction quality.

leads to the most visually stable reconstruction among attacks, indicating its wide applicability in resisting both mask perturbations and pixel-space (RGB) adversarial noise. Against `MaskAttack`, all defense techniques lead to more stable reconstructions compared to the undefended model. `MaskAttack`-MFA exhibits less destructive effect against more robust models, like `RT+TIR` and `SSR`. `MaskAttack`-EM, on the other hand, successfully manages to perturb both `RT+TIR` and `SSR`, indicating that while some defenses lead to robustness against mean noise fluctuations, they still remain susceptible to error maximization attacks. The performance under pixel-based attacks, MFA, and AdvDM, also supports this observation, with AdvDM leading to increased variability during reconstruction (see also Table 1). Finally, it is worth noting that simple random changes on the mask (denoted by `Rand`) still lead to meaningful reconstruction variety, compared to the `No Attack` scenario, indicating a pitfall in standard training paradigms and models.

A natural question arising from Table 1 is: *Why do random mask changes inflict significant degradation?* We identify that the embedding space spanned by standard encoders $\mathcal{E}$ is highly susceptible to small input changes, as induced by the binary pixel flips in the mask. This, in turn, leads to significant distortions over the latent conditioning vector z_c, driving the backward process towards unrealistic reconstructions. A brief discussion on the robustness of $\mathcal{E}$ and $\mathcal{D}$ is included in Appendix D.1.1, providing additional evidence of $\mathcal{E}$'s sensitivity, while a more exhaustive treatment of this matter is left for future work.

Attacks on pre-trained LDM/Stable Diffusion models. To further our study, we test the performance of our `MaskAttack` against an extensively pre-trained LDM model (Rombach et al. 2022), and on two Stable Diffusion checkpoints

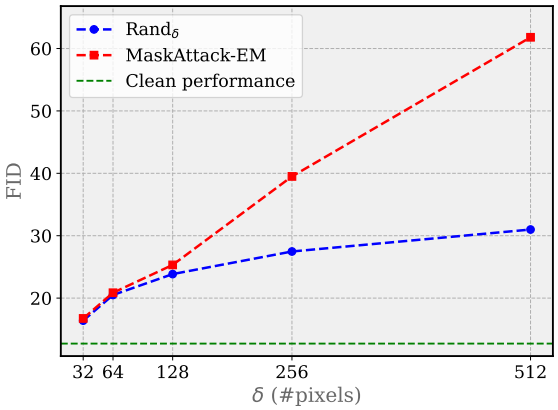


Fig. 5: Attack performance vs. budget on AFHQ-Cats.

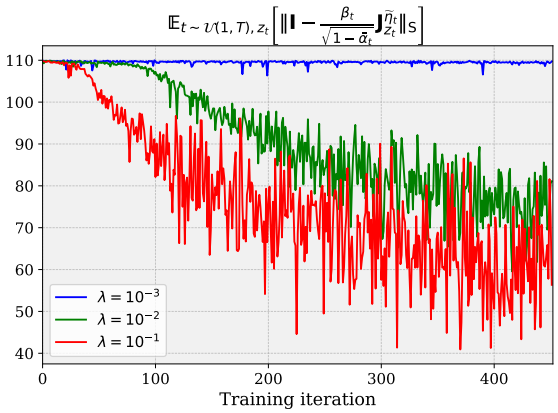


Fig. 6: Influence of different λ on training dynamics, during our proposed sensitivity regularization.

(v1.5 and v2.0), on the Places365 dataset. Given the 512×512 resolution of images contained within this dataset, we use a higher perturbation budget of $\delta = 1024$ mask pixels, allowing less than $\frac{1024}{512^2} \cdot 100 \approx 0.4\%$ mask modifications. Table 3 illustrates the attack performance on these models, with `MaskAttack` outperforming MFA and AdvDM and proving the model's increased susceptibility to mask changes even in higher-resolution settings, regardless of the diffusion model.

5.3 Defenses performance

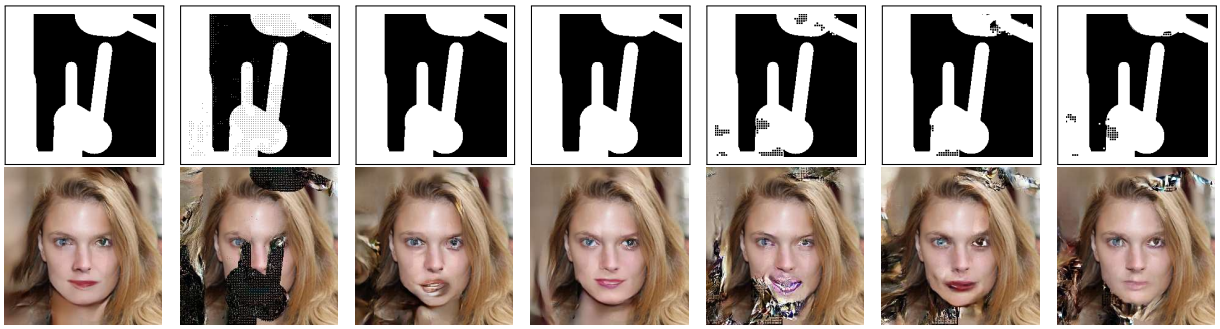
Shifting from an attacker's perspective, in the following, we discuss to what extent defense techniques manage to increase the resilience against

Table 3: Attack performance against pre-trained LDM and SD models on Places365 (512×512) dataset.

Model	No attack		**MFA**		**AdvDM**		`MaskAttack-MFA`		`MaskAttack-EM`		`Rand`$_\delta$	
	FID ↓	LPIPS ↓	FID ↑	LPIPS ↑	FID ↑	LPIPS ↑	FID ↑	LPIPS ↑	FID ↑	LPIPS ↑	FID ↑	LPIPS ↑
`LDM-4`	11.83	0.119	40.97	0.214	20.33	0.179	**61.43**	0.259	56.1	0.237	24.80	**0.301**
`SD 1.5`	11.64	0.109	13.92	0.130	16.38	0.158	40.49	0.205	**42.42**	0.208	32.22	**0.218**
`SD 2.0`	11.10	0.106	13.68	0.134	15.38	0.153	46.26	0.223	**49.92**	**0.227**	25.85	0.215

Table 4: Robustness comparison between `SSR` and *SmoothDiffusion* on CelebA-HQ.

Regularizer	**Clean**		`MaskAttack`	
	FID	LPIPS	FID	LPIPS
SmoothDiffusion($\lambda = 10^{-2}$)	21.15	0.121	60.42	0.151
SmoothDiffusion($\lambda = 10^{-3}$)	15.31	0.113	60.81	0.151
`SSR`($\lambda = 10^{-2}$)	**11.35**	**0.103**	**37.06**	**0.137**
`SSR`($\lambda = 10^{-3}$)	12.05	0.105	57.77	0.148



(a) No attack (b) $\mathcal{F} = \text{Id}$ $\lambda_{\text{TV}} = 0$ (c) $\mathcal{F} = \text{Id}$ $\lambda_{\text{TV}} = 10$ (d) $\mathcal{F} = \text{Id}$ $\lambda_{\text{TV}} = 100$ (e) $\mathcal{F} = \text{SMF}$ $\lambda_{\text{TV}} = 0$ (f) $\mathcal{F} = \text{SMF}$ $\lambda_{\text{TV}} = 10$ (g) $\mathcal{F} = \text{SMF}$ $\lambda_{\text{TV}} = 100$

Fig. 7: Adversarial mask comparison on CelebA-HQ, under different configurations for `MaskAttack-EM`. A budget of $\delta = 1024$ mask pixels is used for better visualization. Top row: inpainting mask. Bottom row: inpainting result.

attacks. Table 1 results indicate that `RT` is sufficient to mitigate the effect of random mask changes, enhancing an LDM's generalization to account for more diverse masks. However, `RT` leads to little-to-no robustness gain against more powerful, gradient-based attackers, indicating that it suffers from robust overfitting. `TIR`, on the other hand, leads to significant robustness gains against our attacker, with, however, a higher susceptibility to random changes than `RT`. Combining these two approaches (`RT+TIR`) aggregates the benefits from each, adding resilience to both random changes and adversarial perturbations.

Our `SSR` regularization method achieves the highest robustness against `MaskAttack`, but also against MFA and AdvDM. This is related to its definition in (27), which shows that `SSR` does not explicitly aim towards robustness w.r.t. m or x, but rather towards diminishing $\widetilde{\eta}_t$'s sensitivity w.r.t. z_t. Hence, it pushes ϵ_θ to learn robust input-output mappings within the latent space, irrespective of the perturbation model. Figure 6 illustrates the training dynamics for `SSR` for different λ, where a value too high leads to unstable behavior and ultimately loss in clean performance, while a value too low leads to no robustness gains. Finally, all defense techniques lead to improved

Table 5: Attack performance for different choices of t_{attack} during `MaskAttack`.

Dataset	Metric	t_{attack}						
		$\sim\mathcal{U}(1,T)$	1	200	600	t^* (23)	800	$T-1$
CelebA-HQ	**FID**↑	56.61	40.63	45.81	61.18	**62.80**	61.95	42.41
	LPIPS↑	0.149	0.151	0.144	0.152	**0.154**	0.154	0.143
AFHQ-Cats	**FID**↑	18.71	15.32	16.41	18.90	**20.86**	18.51	16.74
	LPIPS↑	0.165	0.157	0.160	0.168	**0.174**	0.167	0.163
LSUN-Churches	**FID**↑	18.65	16.28	18.99	20.60	**20.73**	20.02	16.17
	LPIPS↑	0.220	**0.233**	0.214	0.223	0.223	0.215	0.213

Table 6: Performance comparison between full-training and fine-tuning with our SSR regularization, starting from the same initial checkpoint.

	Clean			MaskAttack		
	FID	LPIPS	SSIM	FID	LPIPS	SSIM
Fine-Tuning	**10.93**	**0.120**	**0.815**	**16.39**	**0.154**	**0.801**
Full Training	12.00	0.129	0.798	17.45	0.163	0.786

clean performance, achieving higher generalization besides adversarial robustness. Figure 4 shows a concrete example, where increasing λ starts decorrelating the generated features from the visible context, due to extensive regularization, while still providing decent reconstruction performance.

Full-training vs. Fine-tuning with SSR. To assess whether fine-tuning improves or maintains performance when using our SSR regularizer, we present an additional comparison in Table 6. These results indicate that full training with SSR yields degraded clean generation quality compared to fine-tuning, as expected: applying SSR when the entire network is left unfrozen prevents the model from efficiently leveraging the pre-trained representations. Furthermore, the robustness gap – difference between clean and attack performance – remains identical across both settings: ΔFID = 5.46 for fine-tuning, and ΔFID = 5.45 for full training. This confirms that full training offers no meaningful robustness advantage, while leading to poorer clean performance under regularization training.

Comparison with other diffusion regularizations. Although no other regularization methods in literature explicitly address the stability of diffusion models against adversarial perturbations, we compare our `SSR` with *Smooth Diffusion* (Guo et al. 2024) to strengthen our analysis. Recall that smooth diffusion uses a regularization loss of the form:

$$\mathcal{L}_{\text{smooth}} = \lambda\,\mathbb{E}_{t,\Delta z_0}\left[\sqrt{1-\overline{\alpha}_t}\|\mathbf{J}_{\eta_t}^{\widehat{z}_0}\Delta z_0\|_2 - \overline{\mu}\right]^2, \tag{28}$$

where $\overline{\mu}$ is the exponential moving average of the inner term. Table 4 shows the performance in clean and adversarial settings. `SSR` achieves better results in both scenarios, while *Smooth Diffusion* biases the original denoising objective and leads to no benefit in terms of robustness. We attribute this to `SSR`'s step-wise character, enforcing stability over single-step predictions $\widetilde{\eta}_t$ rather than multi-step reconstructions $\widehat{z}_0$.

5.4 Ablation Studies

In the following, we perform several ablation studies to better illustrate the benefits within our proposed attack and defense frameworks.

Perturbation budget. Figure 5, illustrates how the attack performance evolves under increasing budget δ. Compared to random mask changes,

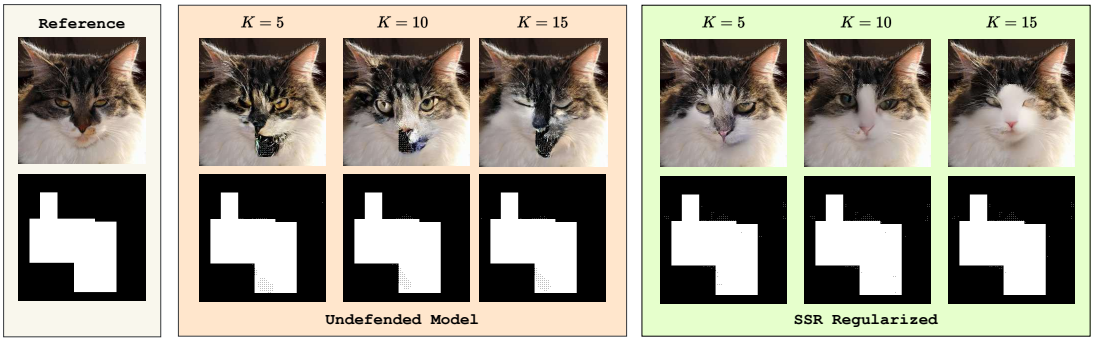


Fig. 8: Comparison between the effect of gradient smoothing steps (K) during `MaskAttack` on an undefended/standard LDM vs. on a `SSR`-regularized LDM.

Table 7: Attack performance on CelebA-HQ (FID↑) under different configurations for `MaskAttack`.

	$\lambda_{\text{TV}} = 0$	$\lambda_{\text{TV}} = 10$	$\lambda_{\text{TV}} = 100$
$\mathcal{F} = \text{Id}$	61.95	17.72	17.46
$\mathcal{F} = \text{SMF}$	**81.14**	51.92	52.64

Table 8: Attack performance (FID) for different gradient smoothing steps K during `MaskAttack` on AFHQ-Cats models.

Defense	$K = 1$	$K = 5$	$K = 10$	$K = 15$	$K = 25$
-	**20.86**	19.29	19.29	19.11	18.72
`TIR`	17.9	18.09	18.48	18.02	**18.53**
`SSR`	16.38	16.73	16.23	**17.11**	16.45

our `MaskAttack` presents an increasing trend, underscoring the importance of our pseudo-ℓ_0 optimization scheme in finding the optimal solution for m_{adv}. Additionally, in Appendix C we provide a theoretical quantification of the maximum pixel-space distortion, given a fixed perturbation budget, thus establishing a connection between encoder-decoder (latent space) sensitivity and the resulting semantic distortion in pixel space.

Attack timestep. We perform an ablation on the timestep t_{attack} at which m_{adv} is optimized. Table 5 provides the attack performance for several choices of t_{attack}, including our proposed t^* from (23). Correlating these numerical results with the illustrated sensitivity $s(t)$ in Figure 1, our theoretical findings are confirmed: attacking too late does not leave room for the perturbation to grow through the generative chain, while attacking too early restricts the perturbation's effect on subsequent timesteps.

Comparison between SMF and TV. We extend our study regarding `MaskAttack`'s configuration by discussing the effect of TV regularization. Table 7 and Figure 7 showcase several results concerning different choices of $(\lambda_{\text{TV}}, \mathcal{F})$.

Increasing λ_{TV} strengthens the smoothness constraint on the generated masks, with this additional regularization generally attenuating their adversarial impact. Using no filter $\mathcal{F} = \text{Id}$ along with TV regularization leads to pixel modifications only along convex 1-valued regions, leading to significantly less powerful attacks, while adopting $\mathcal{F} = \text{SMF}$ allows for more complex mask modifications, even with TV regularization. These findings also indicate that standard inpainting LDMs have an inherent sensitivity to isolated mask changes, being unable to incorporate a rather sparse visible context into surrounding reconstructed masked regions.

Gradient smoothing steps K. Table 8 illustrates the performance of `MaskAttack` for different number of gradient smoothing steps. We observe no significant difference between different values of K, with overall fewer averaging steps actually achieving better attack performance for the undefended model. This shows that a standardly-trained LDM is susceptible enough to simple attack strategies, with the gradient of a local objective having enough power. On the other hand, for our proposed defense mechanisms a higher K leads to more powerful adversarial masks, indicating a smoother loss ($\mathcal{L}_{\text{attack}}$)

Table 9: Performance comparison on CelebA-HQ for different attacks and protective perturbation methods. **Bold** indicates the best performance, while underscore denotes second-best.

	FID↑	LPIPS↑	SSIM↓	PSNR↓
No Attack	16.35	0.112	0.837	19.28
MFA (Yu et al. 2024b)	16.72	0.117	0.835	19.17
AdvDM (Liang et al. 2023)	22.01	0.153	0.807	18.31
DDD (Son et al. 2024)	16.44	0.112	0.836	19.15
DiffVax (Ozden et al. 2025)	16.20	0.113	0.835	19.26
AdvPaint (Jeon et al. 2025)	20.90	0.136	0.818	18.94
PhotoGuard (Salman et al. 2023)	29.46	**0.193**	**0.784**	18.58
MaskAttack-MFA	33.59	0.133	0.823	18.47
MaskAttack-EM	**62.21**	0.153	0.819	**17.03**

landscape and less representative local gradients, motivating the need to average over several realizations at the current mask $m_{\text{adv}}^{(n)}$. Figure 8 shows an example of attack for an undefended and an SSR-regularized LDM, for different gradient smoothing steps K. While for the undefended model, both small and large K result in significant degradation, visually increasing with the number of gradient smoothing steps, the SSR-regularized LDM illustrates significantly higher stability. The profile of adversarial masks differs between the two models, with regularization leading to more sparse changes, indicating less powerful gradient directions during the attack's optimization.

Comparison with protective perturbation methods. We evaluate protective perturbations under an adversarial lens, treating them as attack methods and benchmarking their induced degradation against `MaskAttack`. We employ four well-established immunization methods: PhotoGuard (Salman et al. 2023), DDD (Son et al. 2024), AdvPaint (Jeon et al. 2025), and DiffVax (Ozden et al. 2025). Table 9 shows the performance comparison between our attack and immunization methods. `MaskAttack` leads to the highest degradation in terms of FID and PSNR, demonstrating superior disruption at the distributional level. While the pixel-based EncoderAttack achieves higher LPIPS and lower SSIM, these metrics capture local perceptual and structural discrepancies that are more reflective of pixel-level artifacts than of a genuine failure of the inpainting process.

Table 10: Black-box attack performance against LDM-4 on Places365 dataset.

Setting	FID↑	LPIPS↑	SSIM↓
No Attack	11.83	0.119	0.854
$\texttt{Rand}_\delta$	24.80	0.301	0.787
`LDM-4` → `LDM-4`	61.43	0.259	0.798
`SD 1.5` → `LDM-4`	31.83	0.203	0.816
`SD 2.1` → `LDM-4`	34.14	0.214	0.810

Table 11: Black-box attack performance against *SD 3.0* on Places365 dataset.

Setting	FID↑	LPIPS↑	SSIM↓
No Attack	11.83	0.119	0.854
$\texttt{Rand}_\delta$	14.69	0.131	0.791
`LDM-4` → `SD 3.0`	19.66	0.104	0.815
`SD 1.5` → `SD 3.0`	27.70	0.135	0.776
`SD 2.1` → `SD 3.0`	27.01	0.133	0.778

5.5 Black-box Analysis

To further demonstrate the practical significance of our attack, we analyze its performance within several black-box settings, targeting different open-source pre-trained LDM/SD models. We use the Places365 dataset to evaluate all black-box scenarios, employing the following models: an `LDM-4` architecture pre-trained on Places365, and three Stable Diffusion models – `SD 1.5`, `SD 2.1`, and `SD 3.0`. We aim to answer the following questions:

1. *How well do adversarial masks crafted on larger, presumably stronger models transfer to smaller ones?*
2. *How well do adversarial masks crafted on smaller, presumably weaker models transfer to larger, more powerful ones?*

To answer the first question, in Table 10 we present the black-box transferability results against `LDM-4`, indicating by $\texttt{X} \rightarrow \texttt{Y}$ that adversarial masks are generated on `X` and then tested on `Y`. Results show that `MaskAttack` achieves meaningful black-box transferability: adversarial masks generated on `SD 1.5` and `SD 2.1` result in significantly larger disruptions on `LDM-4` than simple random mask perturbations ($\texttt{Rand}_\delta$). Moreover, as expected, the transferred attacks are less effective than in the white-box setting, consistent with standard observations within the literature.

To address the second question, we analyze the black-box transferability performance on the `SD 3.0` model, which uses a MM-DiT (Esser et al. 2024) architecture instead of the previous UNet-based denoisers. In Table 11, we illustrate the black-box performance, indicating that black-box attacks transferred from models with a fundamentally different architecture (UNet-based) still manage to transfer a more significant adversarial effect than random noise.

Overall, these results highlight the practical threat posed by `MaskAttack` in realistic scenarios where the target model is unknown. Notably, the successful transfer to `SD 3.0` – despite its fundamentally different MM-DiT architecture – underscores the generality of adversarial masks as a perturbation strategy, suggesting that the attack is not tied to a specific architectural family.

5.6 Limitations

Although `MaskAttack` results in significant disruptions in reconstructed areas, in a few cases its performance is surpassed by simple random changes within the mask (see Table 1, in particular the results on SSR-regularized models). This indicates that for robust models, the optimization within Algorithm 1 reaches a weak local optimum, failing to effectively exploit the step-wise gradient signal through the mask. This is also supported by the ablation study regarding gradient smoothing steps, with higher values of K resulting in better performance. We view this as a current limitation of our method. At the same time, it is related to the effectiveness of robust training, which yields less informative gradients for white-box attackers and a smoother loss landscape.

6 Conclusions

In this paper, we investigated the robustness of widely-used latent diffusion models against white-box adversarial attacks. We developed a new strategy named `MaskAttack`, aiming at generating adversarial inpainting masks to maximally perturb backward-reconstructed images. To fully leverage the model sensitivity, we devised an approach for finding the most sensitive timestep within DDPM's backward process, allowing us to optimize adversarial examples. Our experiments on several well-known benchmarks indicate that our attack consistently outperforms previous pixel-based methods, fully accounting for an LDM's susceptibility. Finally, building on our findings, we proposed several techniques to improve robustness, most notably a new regularization technique leveraging our timestep sensitivity measure. By characterizing failure modes and proposing robustness-oriented methods, this article provides a foundation for future work on reliable generative modeling.

Declarations

Funding. This work was supported by the European Union – NextGenerationEU, through the National Recovery and Resilience Plan (PNRR), Component 9, Investment 4, under project SENSE, THINK @ POLITEHNICA BUCUREȘTI (SENTHIPoli) No. 6.PI/I4/C9.

Conflict of interest. Authors declare no conflict of interest.

Data availability. This study does not involve any private or sensitive data. All datasets utilized in the research are obtained from publicly available open-source papers (Xia et al. 2021; Yu et al. 2024a; Choi et al. 2020; Zhou et al. 2017).

A Mean Fluctuation Attack

Algorithm 4 `MFA` attack Yu et al. (2024b), ℓ_∞

Input: Clean image x, mask m, number of attack steps N, perturbation budget δ, gradient step-size α

Output: adversarial image $x_{\text{adv}} = x_{\text{adv}}^{(N)}$

$z_0 = \mathcal{E}(x)$
$x_{\text{adv}}^{(0)} = x + \eta$, $\eta \sim \mathcal{N}(0, \mathrm{I})$
for $n = 0, \ldots, N-1$ **do**
  Compute $z_c^{(n)} = z_c(x_{\text{adv}}^{(n)}, m)$ as in Eq. (3)
  $t_n \sim \mathcal{U}(0, T)$, $\eta^{(n)} \sim \mathcal{N}(0, \mathrm{I})$
  $z^{(n)} = \sqrt{\bar{\alpha}_{t_n}} z_0 + \sqrt{1 - \bar{\alpha}_{t_n}} \eta^{(n)}$
  $\widehat{\eta}^{(n)} = \epsilon_\theta(z^{(n)}, t_n, z_c^{(n)})$
  $L^{(n)} = L(\widehat{\eta}^{(n)})$
  $x_{\text{adv}}^{(n+1)} = \text{proj}_{x+[-\delta,\delta]^{H\times W\times C}}\left(x_{\text{adv}}^{(n)} + \alpha \nabla_x L^{(n)}\right)$
end for

Mean Fluctuation Attack (MFA) (Yu et al. 2024b) is specifically designed for inpainting in latent diffusion models, seeking to determine the adversarial image x_{adv} that causes the greatest output degradation. Algorithm 4 illustrates the MFA attack. It works by updating the clean image x with a ℓ_∞-constrained perturbation to maximize the mean value $L(\widehat{\eta}_t) = \mathsf{average}[\widehat{\eta}_t]$ of the predicted noise $\widehat{\eta}_t$. The mechanism is based on the projected gradient ascent algorithm, with the update direction for x_{adv} given by

$$\begin{aligned}\left(\nabla_x L^{(n)}\right)^\top &= \mathbf{J}^L(\widehat{\eta}^{(n)}) \\ &\quad \times \mathbf{J}_{z_c}^{\widehat{\eta}_t}(z^{(n)}, t_n, z_c^{(n)}) \mathbf{J}_x^{z_c}(x_{\text{adv}}^{(n)}, m), \\ \mathbf{J}_x^{z_c}(x_{\text{adv}}^{(n)}, m) &= \begin{bmatrix} \mathbf{0} \\ \mathbf{J}^{\mathcal{E}}\left((1-m)^{\otimes C} \odot x_{\text{adv}}^{(n)}\right) \mathbf{D}_m \end{bmatrix},\end{aligned} \tag{29}$$

where $\mathbf{J}_y^f$ is the Jacobian of a function f w.r.t vectorized variable y (with no variable specified if f is a function of a single variable), $n \in \mathbb{N}_+$ is the attack step, and $\mathbf{D}_m = \text{Diag}[(1-m)^{\otimes C}]$. One variant of MFA, sets t constant during all attack steps to $\lfloor 4T/5 \rfloor$, resulting from an empirical analysis of the network's sensitivity. Since $\nabla_x L(x_{\text{adv}}^{(n)})$ relies on the encoder, the effectiveness of the attack is heavily influenced by the stability properties of $\mathcal{E}$. However, recent studies examined the robustness of autoencoders by investigating vulnerabilities associated with ill-conditioned layers (Ramanaik et al. 2025), uncovering effective methods to exploit these weaknesses and thereby compromising their reliability.

B Encoder Gradient

Consider the following Jacobian:

$$\begin{aligned}&\mathbf{J}_m^{\widetilde{z}_c}(x, \mathcal{F}(m_{\text{adv}}^{(n)})) = \\ &= \begin{bmatrix} \mathbf{J}^{(\cdot)\downarrow_s}\left(\mathcal{F}(m_{\text{adv}}^{(n)})\right) \\ -\mathbf{J}^{\mathcal{E}}\left((1 - \mathcal{F}(m_{\text{adv}}^{(n)})) \odot x\right) \mathbf{D}_x \end{bmatrix} \mathbf{J}^{\mathcal{F}}(m_{\text{adv}}^{(n)}).\end{aligned} \tag{30}$$

For the Jacobian of encoder $\mathcal{E}$ from (30) we want to show that

$$\mathbb{E}_x[\mathbf{J}^{\mathcal{E}}\left((1 - \mathcal{F}(m_{\text{adv}}^{(n)})) \odot x\right) \mathbf{D}_x] = \mathbf{0}. \tag{31}$$

Denoting $\widetilde{m} = 1 - \mathcal{F}(m_{\text{adv}})$ and assuming x is normalized s.t. $x \sim \mathcal{U}(-1, 1)$, as it is custom in encoder-decoder structures (e.g. VQ-VAE/KL-VAE (Van Den Oord et al. 2017)) and diffusion models, one can see that

$$\mathbb{E}_x[\widetilde{m} \odot x] = \widetilde{m} \odot \mathbb{E}_x[x] = \mathbf{0}, \quad \forall \widetilde{m} \in \{0, 1\}^{H\times W\times C}, \tag{32}$$

so the distribution for the encoder's input is centered. Consider the following random variable:

$$Y = \widetilde{m} \odot X, \;\; X \sim \mathcal{U}(-1, 1). \tag{33}$$

Then, given the symmetry of X, we have, for every $\widetilde{m}$,

$$X \sim -X \Rightarrow \widetilde{m} \odot X \sim -\widetilde{m} \odot X \Leftrightarrow Y \sim -Y. \tag{34}$$

Hence, the encoder's input distribution Y is also symmetric. By making the additional assumption that $\mathcal{E}$ is an odd function, we deduce that $\mathbf{J}^{\mathcal{E}}(\widetilde{m} \odot \cdot)$ is an even function and Eq. (31) holds.

C Sensitivity Analysis of Backward Diffusion

Besides the quantity in (17), one could also be interested in the sensitivity of the final reconstruction z_0 w.r.t. the inpainting mask m.

Considering the backward diffusion in (2) when (4) holds, we can express the Jacobian of the denoised sample at step $t-1$ as follows: for every $t \in \{1, \dots, T\}$,

$$\mathbf{J}_m^{z_{t-1}} = \frac{1}{\sqrt{1-\beta_t}}\Big(\mathbf{J}_m^{z_t} - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}(\mathbf{J}_{z_t}^{\widetilde{\eta}_t}\mathbf{J}_m^{z_t} + \mathbf{J}_{z_c}^{\widetilde{\eta}_t}\mathbf{J}_m^{z_c})\Big), \tag{35}$$

with $\mathbf{J}_m^{z_T} = \mathbf{0}$. The following compact expression of the Jacobian at $t=0$ can be deduced:

$$\mathbf{J}_m^{z_0} = -\left(\sum_{t=1}^{T} \mathbf{\Gamma}_t \mathbf{J}_{z_c}^{\widetilde{\eta}_t}\right)\mathbf{J}_m^{z_c}, \tag{36}$$

with, for every $t \in \{1, \dots, T\}$,

$$\mathbf{\Gamma}_t = \frac{\beta_t}{\sqrt{\alpha_t(1-\bar{\alpha}_t)}}\left(\prod_{i=1}^{t-1}\frac{1}{\sqrt{1-\beta_i}}\Big(\mathbf{I} - \frac{\beta_i}{\sqrt{1-\bar{\alpha}_i}}\mathbf{J}_{z_i}^{\widetilde{\eta}_i}\Big)\right). \tag{37}$$

Proof To prove the above expression, we make use of the following notation:

$$\begin{cases} A_t = \mathbf{J}_m^{z_t} \\ B_t = \frac{1}{\sqrt{1-\beta_t}}\Big(\mathbf{I} - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\mathbf{J}_{z_t}^{\widehat{\eta}_t}\Big) \\ C_t = -\frac{\beta_t}{\sqrt{(1-\beta_t)(1-\bar{\alpha}_t)}}\mathbf{J}_{z_c}^{\widetilde{\eta}_t}\mathbf{J}_m^{z_c}. \end{cases} \tag{38}$$

The iterative diffusion process is then expressed as

$$A_{t-1} = B_t A_t + C_t, \quad t \in \{1, \dots, T\}, \quad A_T = \mathbf{0}. \tag{39}$$

Consider the first terms of this sequence:

$$\begin{aligned} A_T &= \mathbf{0} \\ A_{T-1} &= C_T \\ A_{T-2} &= B_{T-1}A_{T-1} + C_{T-1} \\ &= B_{T-1}C_T + C_{T-1} \\ A_{T-3} &= B_{T-2}A_{T-2} + C_{T-2} \\ &= B_{T-2}B_{T-1}C_T + B_{T-2}C_{T-1} + C_{T-2} \\ &\vdots \end{aligned}$$

By induction, it can be shown that the k-th term with $k \in \{0, \dots, T-1\}$ can be compactly written as

$$A_k = \sum_{t=k+1}^{T}\Big(\big(\prod_{i=k+1}^{t-1} B_i\big)C_t\Big), \tag{40}$$

with the convention $\prod_{i=k+1}^{k}(\cdot) = \mathbf{I}$. In particular,

$$A_0 = \sum_{t=1}^{T}\Big(\big(\prod_{i=1}^{t-1} B_i\big)C_t\Big), \tag{41}$$

and substituting with the original notation, we get the final expression for $\mathbf{J}_m^{z_0}$ in (36).

An upper bound for A_0 could be defined similarly to (21), enabling a direct mask-specific robustness analysis. Since this formulation is overly complex and would result in several implementation challenges, we instead consider in our experiments the more general form in (17), which allows for efficient integration within diffusion training/inference pipelines. □

Relationship between latent space sensitivity and pixel-space distortion. Through the closed-form expression of $\mathbf{J}_m^{z_0}$ from (41) we can quantify an upper bound on the latent space perturbation Δz_0 over the final-step reconstruction z_0, given a mask perturbation vector Δm:

$$\sup \|\Delta z_0\|_2 = \|A_0\|_S \|\Delta m\|_2, \tag{42}$$

where we have used the ℓ_2 norm for both latent space and mask space to create the link to the Jacobian spectral norm. Nonetheless, since $\|\Delta m\|_2 = \sqrt{\|\Delta m\|_0}$ given Δm's binary nature, the two norms are monotonically related, hence the supremum in (42) increases whenever $\|\Delta m\|_0$ increases.

Finally, to link latent space fluctuations Δz_0 with pixel-space perturbations Δx_0, we need to account for the decoder's ($\mathcal{D}$) sensitivity, resulting in the following upper bound:

$$\begin{aligned} \sup \|\Delta \widehat{x}_0\|_2 &= \|\mathbf{J}_{\mathcal{E}(x)}^{\mathcal{D}}\|_2 \sup \|\Delta z_0\|_2 &(43)\\ &= \|\mathbf{J}_{\mathcal{E}(x)}^{\mathcal{D}}\|_2 \|\mathbf{J}_m^{z_0}\|_2 \|\Delta m\|_2. &(44) \end{aligned}$$

The above expression provides a direct link between the perturbation budget of mask m and the maximum pixel-space perturbation over the final reconstruction $\widehat{x}_0$. In Appendix D.1.1 we empirically compute $\|\mathbf{J}_{\mathcal{E}(x)}^{\mathcal{D}}\|_2$ for several architectural choices and datasets, showcasing that the transformation implied by $\mathcal{D}$ is relatively stable, compared to the compression implied by encoder $\mathcal{E}$.

D Models and Training

D.1 Architecture

We experiment with different variants of the original LDM architecture (Rombach et al. 2022), largely utilized in current Stable Diffusion models[1]. At its core, each LDM is comprised of an encoder-decoder compressor $\mathcal{D} \circ \mathcal{E}$, and a denoiser ϵ_θ acting in the compressor's latent space. We use a relatively small scale UNet for training our LDMs on CelebA-HQ and AFHQ-Cats due to their simplicity, while for LSUN-Churches, we scale the inner model accordingly until obtaining decent clear reconstruction performance. The choice of $\mathcal{D} \circ \mathcal{E}$ will be discussed in the next section.

D.1.1 Encoder-decoder choice

Although previous work mostly relied on VQ-VAE models to encode input data, we observe that the standard KL-VAE yields better reconstruction performance (see Table 12). Emphasizing the need for a sufficiently expressive latent space, we choose KL-VAE over VQ-VAE during further experiments regarding generative inpainting, unless stated otherwise.

Inspired by Ramanaik et al. (2025), in order to measure the robustness profile of encoder-decoder structures, we compute statistics for the main Jacobians: $\|\mathbf{J}_x^{\mathcal{E}}\|_2$, $\|\mathbf{J}_{\mathcal{E}}^{\mathcal{D}}\|_2$, and $\|\mathbf{J}_x^{\mathcal{D}\circ\mathcal{E}}\|_2$. Specifically, we compute the mean and standard deviation of the aforementioned quantities using 100 images from the test set. Table 13 shows these statistics ($\mu \pm \sigma$), computed separately for images from each dataset. Jacobian spectral norms are estimated with `deepinv` library[2], using the default settings. The results indicate that, while the mapping implied by decoder $\mathcal{D}$ is relatively stable, regardless of the architecture or the dataset, the encoder $\mathcal{E}$ implies a significant expansion from the original space to the latent space. This could indicate why LDMs are susceptible to image-based attacks such as MFA and AdvDM, given that the encoded latent space represents the guiding condition. We highlight the fact that this could open the door towards increasing the robustness of LDMs by accounting to the robustness properties of the latent space, leaving it as a possible future research endeavor.

[1] https://stability.ai/news/introducing-stable-diffusion-3-5
[2] https://github.com/deepinv/deepinv



(a) Input (b) DDPM (c) DDIM (d) DPM-S

Fig. 9: Reconstruction examples using different samplers. DPM-S stands for DPM-Solver.

D.2 Training

We used the standard linear noise schedule for training our LDMs: $T = 1000, \beta_1 = 0.0015, \beta_T = 0.0205$. The base denoising loss considered is:

$$\mathcal{L}_{\text{base}}(x, m, t) = \|\eta_t - \epsilon_\theta\big(z_t, t, z_c(x, m)\big)\|_2^2. \quad (45)$$

For optimization we used AdamW (Loshchilov et al. 2017) with a base learning rate $\texttt{lr} = 2e-5$ and a warmup cosine scheduler. The inpainting mask for each training image x was generated on-the-fly during each training step, with the goal of increasing generalization.

D.3 Inference

We compare three different samplers: DDPM, DDIM (Song et al. 2020), and DPM-Solver (Lu et al. 2022). Table 14 shows their reconstruction performance on AFHQ-Cats images, with some examples shown in Figure 9. Since our study prioritizes a robustness analysis rather than achieving state-of-the-art performance, we adopted DPM-Solver in our next experiments for efficiency purposes only.

D.4 Attacks Configuration

For MFA (Yu et al. 2024b), we use a perturbation budget of $\delta = \frac{8}{255}$ for all RGB input images, in line with the attack configuration suggested by the authors. We use a stepsize of $\alpha = \frac{2}{255}$ and

Table 12: Encoder-Decoder reconstruction performance.

Dataset	Model	FID ↓	LPIPS ↓	SSIM↑
AFHQ-Cats	VQ-VAE $\downarrow_4$	25.29	0.179	0.802
	KL-VAE $\downarrow_4$	5.07	0.028	0.858
CelebA-HQ	VQ-VAE $\downarrow_4$	31.53	0.086	0.863
	KL-VAE $\downarrow_4$	3.38	0.015	0.912

Table 13: Encoder-Decoder robustness evaluation.

Dataset	Model	$\|\mathbf{J}_x^{\mathcal{E}}\|_2$	$\|\mathbf{J}_{\mathcal{E}(x)}^{\mathcal{D}}\|_2$	$\|\mathbf{J}_x^{\mathcal{D}\circ\mathcal{E}}\|_2$
AFHQ-Cats	VQ-VAE $\downarrow_4$	323.26 ± 189.43	6.57 ± 1.97	155.57 ± 71.55
	KL-VAE $\downarrow_4$	8069.50 ± 5460.82	5.05 ± 2.31	2455.05 ± 1675.51
CelebA-HQ	VQ-VAE $\downarrow_4$	405.04 ± 215.09	6.60 ± 2.55	166.22 ± 74.31
	KL-VAE $\downarrow_4$	9202.43 ± 6094.97	4.00 ± 1.62	1815.31 ± 971.49

Table 14: Sampler comparison on AFHQ-Cats test set.

Sampler	NFEs	FID ↓	LPIPS ↓	SSIM↑
DDPM	1000	10.57	0.110	0.814
DDIM	50	12.03	0.124	0.806
DPM-Solver	10	12.71	0.133	0.796

$N = 50$ steps. Following their empirical findings, we set $t_n = 800$ throughout all iterations, as this was shown to result in the highest degradation (also confirmed by Yu et al. (2024b)). A detailed algorithm for MFA can be found in Appendix A.

For AdvDM (Liang et al. 2023), we use similar hyperparameters as for MFA, for a fair comparison, with the corresponding attack maximization loss.

For our MaskAttack, we use $\delta = 64$ for CelebA-HQ and AFHQ-Cats, and $\delta = 128$ for LSUN-Churches. We found that a lower δ for LDMs trained on LSUN-Churches did not achieve optimal results, which could be attributed to the fact that the architectures of ϵ_θ adopted for LSUN-Churches are significantly scaled to account for the dataset's complexity. Nonetheless, we stress the fact that our attack allows only for a relatively small percentage of pixels to be flipped, with less than $\frac{64}{256^2} \cdot 100 \approx 0.1\%$ of total mask pixels for CelebA-HQ and AFHQ-Cats, and less than 0.2% for LSUN-Churches. By contrast, MFA and AdvDM have no restrictions on the number of modified pixels, since they operate on ℓ_∞ budgets δ, hence any pixel in the original image can be modified by $\pm\delta$.

We use $N = 10$ steps, which we found sufficient to obtain good performance, making our attack more resource efficient than image-based attacks, where the number of steps is significantly higher due to the complexity of working within a continuous image domain. Budget δ is uniformly partitioned into step-wise budgets $\delta_n = \lfloor \delta/N \rfloor$, as we have found other methods to achieve lower attack performance. To obtain an estimate for the most sensitive timestep t^* (23), we estimate $s(t)$ according to the description of Figure 1, adopting the timestep which results in the highest $\mu(t) + \sigma(t)$. The t corresponding to the highest peak in Figure 1 is taken as the attack optimization timestep.

D.5 Defenses Configuration

`RT.` For randomized mask training, we adopt a probability of $p = 0.75$ of applying the random perturbation, and a number of random mask changes of $\delta = 64$. We have tested with several values for $p \in \{0.25, 0.5, 0.75\}$, finally sticking with the value which led to the best performance against random mask changes $\texttt{Rand}_\delta$.

`TIR`. In order to generate transformed masks $m_p = \mathcal{T}_p(m)$ with a high degree of diversity, relative to the original mask m, we chose to uniformly sample $p \sim \mathcal{U}(0.01, 0.1)$ at each training step, i.e. a minimum and maximum overlap of 1% and 10%, respectively. We use a factor $\tau = 1.0$ for our regularization term $\mathcal{L}_c$, giving it the same contribution as the original MSE loss terms. In the `RT + TIR` settings, first we apply the random mask transform then we use the $\mathcal{T}_p$ operator to create m_p.

`SSR`. For each scenario, we tested multiple values for λ in order to achieve a balance between good performance on clean data and adversarial robustness. For models trained on AFHQ-Cats and CelebA-HQ we used $\lambda = 10^{-2}$, and for LUSN-Churches a $\lambda = 2 \cdot 10^{-3}$. In an effort to avoid a full training cycle where, at each optimization step, a Jacobian spectral norm term is to be computed (see Eq. 27), we adopt a fine-tuning approach for training our `SSR`-regularized models, using the checkpoint of undefended LDMs as initialization. We have tested several fine-tuning methods, in order to allow the model to retain its initial performance while still leaving room for regularization to take effect. In the end, we adopted a decoder-only fine-tuning approach for ϵ_θ, fixing the encoder part of UNet and training only the decoding blocks. Inspired by cross-attention fine-tuning in Gheini et al. (2021), where only attention layers within decoding blocks are left unfreezed in order to adapt transformers for machine translation, we take a similar approach by allowing the entire decoding part to be trainable. This is motivated by the inpainting context, with only the visible pixels serving as guidance, and to allow more flexibility during regularization.

References

Anciukevičius T, Xu Z, Fisher M, Henderson P, Bilen H, Mitra NJ, et al. Renderdiffusion: Image diffusion for 3d reconstruction, inpainting and generation. In: Proc. of Conf. on Computer Vision and Pattern Recognition; 2023. p. 12608–12618.

Bastounis A, Hansen AC, Vlačić V. The mathematics of adversarial attacks in AI–Why deep learning is unstable despite the existence of stable neural networks. preprint arXiv:210906098. 2021;.

Betti F, Baraldi L, Cucchiara R, Sebe N. Hallucination Early Detection in Diffusion Models. International Journal of Computer Vision. 2026;134(1):35.

Büsing C, Gersing T, Koster AM. A branch and bound algorithm for robust binary optimization with budget uncertainty. Mathematical Programming Computation. 2023;15(2):269–326.

Cao Z, He Y, Liu A, Xie J, Wang Z, Chen F. Cofi-dec: Hallucination-resistant decoding via coarse-to-fine generative feedback in large vision-language models. In: Proc. of International Conference on Multimedia; 2025. p. 10709–10718.

Chen H, Zhang Y, Cun X, Xia M, Wang X, Weng C, et al. Videocrafter2: Overcoming data limitations for high-quality video diffusion models. In: Proc. of Conf. on Computer Vision and Pattern Recognition; 2024. p. 7310–7320.

Chen Y, Viégas FB, Wattenberg M. Beyond Surface Statistics: Scene Representations in a Latent Diffusion Model. CoRR. 2023;.

Chen Y, Chen J, Pan Y, Li Y, Yao T, Chen Z, et al. Improving text-guided object inpainting with semantic pre-inpainting. In: Proc. of European Conference on Computer Vision; 2024. p. 110–126.

Cheng R, Ding Y, Cao S, Duan R, Jia X, Yuan S, et al. Pbi-attack: Prior-guided bimodal interactive black-box jailbreak attack for toxicity maximization. In: Proc. of Conf. on Empirical Methods in Natural Language Processing; 2025. p. 609–628.

Choi JS, Lee K, Jeong J, Xie S, Shin J, Lee K. DiffusionGuard: A Robust Defense Against Malicious Diffusion-based Image Editing. In: International Conference on Learning Representations, ICLR 2025 International Conference on Learning Representations; 2025. .

Choi Y, Uh Y, Yoo J, Ha JW. StarGAN v2: Diverse Image Synthesis for Multiple Domains. In: Proc. of Conf. on Computer Vision and

Pattern Recognition; 2020. .

Corneanu C, Gadde R, Martinez AM. Latentpaint: Image inpainting in latent space with diffusion models. In: Proc. Winter Conference on Applications of Computer Vision; 2024. p. 4334–4343.

Dai X, Liang K, Xiao B. AdvDiff: Generating Unrestricted Adversarial Examples using Diffusion Models. preprint arXiv:230712499. 2023;.

Du C, Li Y, Qiu Z, Xu C. Stable diffusion is unstable. Proc of Advances in Neural Information Processing Systems. 2023;36:58648–58669.

Esser P, Kulal S, Blattmann A, Entezari R, Müller J, Saini H, et al. Scaling rectified flow transformers for high-resolution image synthesis. In: Forty-first international conference on machine learning; 2024. .

Fotiadis S, Valencia ML, Hu S, Garasto S, Cantwell CD, Bharath AA. Disentangled generative models for robust prediction of system dynamics. In: Proc. International Conference on Machine Learning; 2023. p. 10222–10248.

Gheini M, Ren X, May J. Cross-Attention is All You Need: Adapting Pretrained Transformers for Machine Translation. In: Proc. Conf. on Empirical Methods in Natural Language Processing; 2021. p. 1754–1765.

Gong S, Zhang R, Zheng H, Gu J, Jaitly N, Kong L, et al. DiffuCoder: Understanding and Improving Masked Diffusion Models for Code Generation. preprint arXiv:250620639. 2025;.

Guo J, Xu X, Pu Y, Ni Z, Wang C, Vasu M, et al. Smooth diffusion: Crafting smooth latent spaces in diffusion models. In: Proc. of Conf. on Computer Vision and Pattern Recognition; 2024. p. 7548–7558.

Ho J, Jain A, Abbeel P. Denoising diffusion probabilistic models. Proc of Advances in Neural Information Processing Systems. 2020;33:6840–6851.

Hong S, Lee G, Jang W, Kim S. Improving sample quality of diffusion models using self-attention guidance. In: Proc. of Conf. on Computer Vision and Pattern Recognition; 2023. p. 7462–7471.

Huang R, Huang J, Yang D, Ren Y, Liu L, Li M, et al. Make-an-audio: Text-to-audio generation with prompt-enhanced diffusion models. In: Proc. of International Conference on Machine Learning PMLR; 2023. p. 13916–13932.

Huang Z, Wu T, Jiang Y, Chan KC, Liu Z. Reversion: Diffusion-based relation inversion from images. In: SIGGRAPH Asia Conference Papers; 2024. p. 1–11.

Jeon J, Kim WJ, Ha S, Son S, Yoon SE. AdvPaint: Protecting Images from Inpainting Manipulation via Adversarial Attention Disruption. In: 13th International Conference on Learning Representations The International Conference on Learning Representations; 2025. .

Kang M, Song D, Li B. Diffattack: Evasion attacks against diffusion-based adversarial purification. Proc of Advances in Neural Information Processing Systems. 2023;36:73919–73942.

Khader F, Müller-Franzes G, Tayebi Arasteh S, Han T, Haarburger C, Schulze-Hagen M, et al. Denoising diffusion probabilistic models for 3D medical image generation. Nature Scientific Reports. 2023;13(1):7303.

Kolmogorov V, Rother C. Minimizing nonsubmodular functions with graph cuts-a review. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2007;29(7):1274–1279.

Li Y, Chen Z, Jin K, Wang J, Lei J, Li B, et al. Consistency purification: Effective and efficient diffusion purification towards certified robustness. Proc of Advances in Neural Information Processing Systems. 2024;37:105017–105038.

Li Z, Li Z, Cui Z, Pollefeys M, Oswald MR. Sat2scene: 3D urban scene generation from satellite images with diffusion. In: Proc. of Conf. on Computer Vision and Pattern Recognition; 2024. p. 7141–7150.

Liang C, Wu X, Hua Y, Zhang J, Xue Y, Song T, et al. Adversarial example does good: Preventing painting imitation from diffusion models via adversarial examples. preprint arXiv:230204578. 2023;.

Lin S, Liu B, Li J, Yang X. Common diffusion noise schedules and sample steps are flawed. In: Proc. of Winter Conf. on Applications of Computer Vision; 2024. p. 5404–5411.

Liu H, Xue W, Chen Y, Chen D, Zhao X, Wang K, et al. A Survey on Hallucination in Large Vision-Language Models. CoRR. 2024;.

Liu H, Chen Z, Yuan Y, Mei X, Liu X, Mandic D, et al. Audioldm: Text-to-audio generation with latent diffusion models. preprint arXiv:230112503. 2023;.

Loshchilov I, Hutter F, et al. Fixing weight decay regularization in adam. preprint arXiv:171105101. 2017;5(5):5.

Lu C, Zhou Y, Bao F, Chen J, Li C, Zhu J. Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps. Proc of Advances in Neural Information Processing Systems. 2022;35:5775–5787.

Lugmayr A, Danelljan M, Romero A, Yu F, Timofte R, Van Gool L. Repaint: Inpainting using denoising diffusion probabilistic models. In: Proc. of Conf. on Computer Vision and Pattern Recognition; 2022. p. 11461–11471.

Ma J, Li Y, Xiao Z, Cao A, Zhang J, Ye C, et al. Jailbreaking prompt attack: A controllable adversarial attack against diffusion models. In: Findings of the Association for Computational Linguistics: NAACL; 2025. p. 3141–3157.

Nie W, Guo B, Huang Y, Xiao C, Vahdat A, Anandkumar A. Diffusion Models for Adversarial Purification. In: Proc. International Conference on Machine Learning; 2022. p. 16805–16827.

Ozden TC, Kara O, Akcin O, Zaman K, Srivastava S, Chinchali SP, et al. DiffVax: Optimization-Free Image Immunization Against Diffusion-Based Editing. In: The Fourteenth International Conference on Learning Representations; 2025. .

Qi X, Huang K, Panda A, Henderson P, Wang M, Mittal P. Visual adversarial examples jailbreak aligned large language models. In: Proceedings of the AAAI conference on artificial intelligence, vol. 38; 2024. p. 21527–21536.

Ramanaik CK, Roy A, Ntoutsi E. ALMA: Aggregated Lipschitz Maximization Attack on Auto-encoders. preprint arXiv:250503646. 2025;.

Rombach R, Blattmann A, Lorenz D, Esser P, Ommer B. High-resolution image synthesis with latent diffusion models. In: Proc. of Conf. on Computer Vision and Pattern Recognition; 2022. p. 10684–10695.

Rosaria BM, Mirza MH, Lisanti G, Masi I. What is Adversarial Training for Diffusion Models? preprint arXiv:250521742. 2025;.

Salman H, Khaddaj A, Leclerc G, Ilyas A, Madry A. Raising the Cost of Malicious AI-Powered Image Editing. In: International Conference on Machine Learning PMLR; 2023. p. 29894–29918.

Schramowski P, Brack M, Deiseroth B, Kersting K. Safe latent diffusion: Mitigating inappropriate degeneration in diffusion models. In: Proc. of Conf. on Computer Vision and Pattern Recognition; 2023. p. 22522–22531.

Shafahi A, Najibi M, Ghiasi MA, Xu Z, Dickerson J, Studer C, et al. Adversarial training for free! Proc of Advances in Neural Information Processing Systems. 2019;32.

Singh M, Cambronero J, Gulwani S, Le V, Negreanu C, Verbruggen G. Codefusion: A pre-trained diffusion model for code generation. In: Proc. of Conf. on Empirical Methods in Natural Language Processing; 2023. p. 11697–11708.

Son G, Lee J, Woo SS. Disrupting diffusion-based inpainters with semantic digression. In: Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence; 2024. p. 7735–7743.

Song J, Meng C, Ermon S. Denoising diffusion implicit models. Proc of International Conf on Learning Representations. 2020;.

Song Y, Sohl-Dickstein J, Kingma DP, Kumar A, Ermon S, Poole B. Score-Based Generative Modeling through Stochastic Differential Equations. In: Proc. of International Conference on Learning Representations; 2021. .

Suvorov R, Logacheva E, Mashikhin A, Remizova A, Ashukha A, Silvestrov A, et al. Resolution-robust large mask inpainting with fourier convolutions. In: Proc. of the Winter Conf. on Applications of Computer Vision; 2022. p. 2149–2159.

Tang L, Zhang L. Robust overfitting does matter: Test-time adversarial purification with FGSM. In: Proc. of Conf. on Computer Vision and Pattern Recognition; 2024. p. 24347–24356.

Van Den Oord A, Vinyals O, et al. Neural discrete representation learning. Advances in neural information processing systems. 2017;30.

Vice J, Akhtar N, Shah M, Hartley R, Mian AS. Safety Without Semantic Disruptions: Editing-free Safe Image Generation via Context-preserving Dual Latent Reconstruction. In: Proc. of Conf. on Computer Vision and Pattern Recognition; 2025. p. 2306–2316.

Wei A, Haghtalab N, Steinhardt J. Jailbroken: How Does LLM Safety Training Fail? preprint arXiv:230702483. 2023;.

Wu H, Zhang M, He L, Zhou JZ, Lv J. Enhancing Diffusion Model Stability for Image Restoration via Gradient Management. In: Proc. of International Conference on Multimedia; 2025. p. 10768–10777.

Xia W, Yang Y, Xue JH, Wu B. TediGAN: Text-Guided Diverse Face Image Generation and Manipulation. In: Prof. of Conf. on Computer Vision and Pattern Recognition; 2021. .

Yin H, Si G, Wang Z. ClearSight: Visual Signal Enhancement for Object Hallucination Mitigation in Multimodal Large Language Models. In: Proc. of the Computer Vision and Pattern Recognition Conference; 2025. p. 14625–14634.

Yu F, Zhang Y, Song S, Seff A, Xiao J.: LSUN Churches. TIB; 2024. https://service.tib.eu/ldmservice/dataset/lsun-churches.

Yu H, Chen J, Ding X, Zhang Y, Tang T, Ma H. Step vulnerability guided mean fluctuation adversarial attack against conditional diffusion models. In: Proc. of AAAI Conf. on Artificial Intelligence, vol. 38; 2024. p. 6791–6799.

Zhang C, Hu M, Li W, Wang L. Adversarial attacks and defenses on text-to-image diffusion models: A survey. Information Fusion. 2025;114:102701.

Zhao F, Li R, Liu X, Xu L. Soft-Median Choice: An Automatic Feature Smoothing Method for Sound Event Detection. preprint arXiv:201112564. 2020;.

Zhao Y, Barnes C, Zhou Y, Shechtman E, Amirghodsi S, Fowlkes C. Geofill: Reference-based image inpainting with better geometric understanding. In: Proc. of Winter Conference on Applications of Computer Vision; 2023. p. 1776–1786.

Zhou B, Lapedriza A, Khosla A, Oliva A, Torralba A. Places: A 10 million Image Database for Scene Recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2017;40(6):1452–1464.

Zou A, Wang Z, Kolter JZ, Fredrikson M. Universal and Transferable Adversarial Attacks on Aligned Language Models. preprint arXiv:230715043. 2023;.