Pixel Is Not A Barrier: An Effective Evasion Attack for Pixel-Domain Diffusion Models

Chun-Yen Shih[1, 3*], **Li-Xuan Peng**[3*], **Jia-Wei Liao**[1, 3], **Ernie Chu**[2, 3 †], **Cheng-Fu Chou**[1], **Jun-Cheng Chen**[3]

[1] National Taiwan University,
[2] Johns Hopkins University,
[3] Research Center for Information Technology Innovation, Academia Sinica

Abstract

Diffusion Models have emerged as powerful generative models for high-quality image synthesis, with many subsequent image editing techniques based on them. However, the ease of text-based image editing introduces significant risks, such as malicious editing for scams or intellectual property infringement. Previous works have attempted to safeguard images from diffusion-based editing by adding imperceptible perturbations. These methods are costly and specifically target prevalent Latent Diffusion Models (LDMs), while Pixel-domain Diffusion Models (PDMs) remain largely unexplored and robust against such attacks. Our work addresses this gap by proposing a novel attack framework, AtkPDM. AtkPDM is mainly composed of a feature representation attacking loss that exploits vulnerabilities in denoising UNets and a latent optimization strategy to enhance the naturalness of adversarial images. Extensive experiments demonstrate the effectiveness of our approach in attacking dominant PDM-based editing methods (e.g., SDEdit) while maintaining reasonable fidelity and robustness against common defense methods. Additionally, our framework is extensible to LDMs, achieving comparable performance to existing approaches.

Project page — https://alexpeng517.github.io/AtkPDM/

Figure 1: Overview of our attack scenario. Diffusion-based image editing can generate high-quality image variation based on the clean input image. However, by adding carefully crafted perturbation to the clean image, the diffusion process will be disrupted, producing a corrupted image or unrelated image semantics to the original image.

1 Introduction

In recent years, Generative Diffusion Models (GDMs) (Ho, Jain, and Abbeel 2020; Song, Meng, and Ermon 2021) emerged as powerful generative models that can produce high-quality images, propelling advancements in image editing and artistic creations. The *ease* of using these models to edit (Meng et al. 2021; Wang, Zhao, and Xing 2023; Zhang et al. 2023) or synthesize new images (Dhariwal and Nichol 2021; Rombach et al. 2022) has raised concerns about potential malicious usage and intellectual property infringement. For example, malicious users could effortlessly craft fake images with someone's identity or mimic the style of a specific artist. An effective protection against these threats is to craft an adversarial image to force the diffusion model to generate corrupted images or unrelated images to the original inputs. Researchers have made significant strides in crafting imperceptible adversarial perturbations on images to protect against diffusion-based editing.

Previous works such as PhotoGuard (Salman et al. 2023) and Glaze (Shan et al. 2023) have effectively attacked Latent Diffusion Models (LDMs) by minimizing the latent distance between the protected images and their target counterparts. PhotoGuard first introduces attacking either encoders or diffusion process in LDMs via Projected Gradient Descent (PGD) (Madry et al. 2018) for the protection purpose; however, it requires backpropagating the entire diffusion process, making it prohibitively expensive. Subsequent works AdvDM (Liang et al. 2023) and Mist (Liang and Wu 2023) leverage the semantic loss and textural loss combined with Monte Carlo method to craft adversarial images both effectively and efficiently. Diff-Protect (Xue et al. 2024) further improve adversarial effectiveness and optimization

[*] Equal contribution.
[†] Work done as research assistant at CITI, Academia Sinica.

speed via Score Distillation Sampling (SDS) (Poole et al. 2023), setting the state-of-the-art performance on LDMs.

However, previous works primarily focus on LDMs, and attacks on Pixel-domain Diffusion Models (PDMs) remain unexplored. Xue et al. (Xue et al. 2024) also highlighted a critical limitation of current methods: the attacking effectiveness is mainly attributed to the vulnerability of the VAE encoders in LDM; however, PDMs don't have such encoders, making current methods hard to transfer to PDMs. The latest work (Xue and Chen 2024) has attempted to attack PDMs, but the result suggests that PDMs are robust to pixel-domain perturbations. Our goal is to mitigate the gap between these limitations.

In this paper, we propose an innovative framework AtkPDM, to effectively attack PDMs. Our approach includes a novel **feature attacking loss** that exploits the vulnerabilities in denoising UNet to distract the model from recognizing the correct semantics of the image, a **fidelity loss** that acts as optimization constraints that ensure the imperceptibility of adversarial image and controls the attack budget, and a **latent optimization strategy** utilizing victim-model-agnostic VAEs to further enhance the naturalness of our adversarial image. With extensive experiments on different PDMs, the results show that our method is effective and affordable while robust to prevalent defense methods and exhibiting attack transferability in the black-box setting. In addition, our approach outperforms current semantic-loss-based and PGD-based methods, reaching state-of-the-art performance on attacking PDMs. Our contributions are summarized as follows:

1. We propose a novel attack framework targeting PDMs, achieving state-of-the-art performance in safeguarding images from being edited by SDEdit.
2. We propose a novel feature attacking loss design to distract UNet feature representation effectively.
3. We propose a latent optimization strategy via model-agnostic VAEs to enhance the naturalness of our adversarial images.

2 Related Work

2.1 Image Editing with SDEdit-based Methods

With the multi-step sampling nature and the ease of converting a sample to intermediate noisy latent via forward diffusion of Diffusion Models (Ho, Jain, and Abbeel 2020). SDEdit (Meng et al. 2021) indicates that the diffusion model sampling process is not necessarily required to begin with random Gaussian noise, but allows starting with a mixture of input image and noise at arbitrary strength, i.e. forwarded to $t \in [0, T]$, for the editing. This technique is generalized to both PDMs and LDMs. Subsequent editing frameworks (Hertz et al. 2023; Tumanyan et al. 2023; Parmar et al. 2023; Mokady et al. 2023) also build upon this concept.

2.2 Evasion Attack for Diffusion Model

To counteract SDEdit-based editing, Salman et al. first proposed PhotoGuard (Salman et al. 2023) to introduce two attacking paradigms based on Projected Gradient Descent (PGD) (Madry et al. 2018). The first is the Encoder Attack, which aims to disrupt the latent representations of the Variational Autoencoder (VAE) of the LDMs, and the second is the Diffusion Attack, which focuses more on disrupting the entire diffusion process of the LDMs. The Encoder Attack is simple yet effective, but the attacking results are sub-optimal due to its less flexibility for optimization than the Diffusion Attack. Although the Diffusion Attack achieves better attack results, it is prohibitively expensive due to its requirement of backpropagation through all the diffusion steps. In the following, we introduce other proposed method targeting different aspects for attacking diffusion models.

Diffusion Attacks. Despite the cost of performing the Diffusion Attack, the higher generalizability and universally applicable nature drive previous works focusing on disrupting the process with lower cost. Liang et al. (Liang et al. 2023) proposed AdvDM to utilize the diffusion training loss as their attacking semantic loss. Then, AdvDM performs gradient ascent with the Monte Carlo method, aiming to disrupt the denoising process without calculating full backpropagation. Mist (Liang and Wu 2023) also incorporates semantic loss and performs constrained optimization via PGD to achieve better attacking performance.

Encoder Attacks. On the other hand, researchers found that VAEs in widely adopted LDMs are more vulnerable to attack at a lower cost than the expensive diffusion process. Hence, they (Salman et al. 2023; Liang and Wu 2023; Shan et al. 2023; Xue et al. 2023) focus on disrupting the latent representation in LDM via PGD and highlight the encoder attacks are more effective against LDMs.

Conditional Module Attacks. Most of the LDMs contain conditional modules for steering generation, previous works (Shan et al. 2023, 2024; Lo et al. 2024) exploited the vulnerability of text conditioning modules. By disrupting the cross-attention between text concepts and image semantics, these methods effectively interfere with the diffusion model's ability to capture image-text alignment, thereby achieving the attack.

Limitations of Current Methods. To the best of our knowledge, previous works primarily focus on adversarial attacks for LDMs, while attacks on PDMs remain unexplored. Xue et al. (Xue and Chen 2024) further emphasized the difficulty of attacking PDMs. However, in our work, we find that by crafting an adversarial image to corrupt the intermediate representation of diffusion UNet, we can achieve promising attack performance for PDMs, while the attack is also compatible with LDMs. Moreover, inspired by (Laidlaw, Singla, and Feizi 2021; Liu et al. 2023) which utilize LPIPS (Zhang et al. 2018) as the distortion measure, we also propose a novel attacking loss as the measure to craft better adversarial images for PDMs.

3 Methodology

3.1 Threat Model and Problem Setting

The malicious user collects an image $\mathbf{x}$ from the internet and uses SDEdit (Meng et al. 2021) to generate unautho-

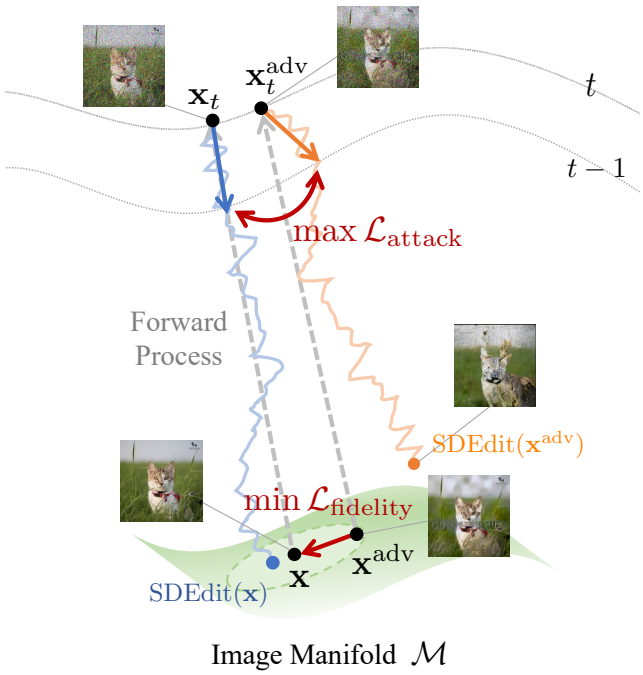

Figure 2: Conceptual illustration of our method. We randomly forward both the clean image $\mathbf{x}$ and adversarial image $\mathbf{x}^{\text{adv}}$ to noise level t, then utilize our feature attacking loss to maximize the feature distance between noisy latent $\mathbf{x}_t$ and $\mathbf{x}_t^{\text{adv}}$ in the reverse process of diffusion models while imposing our fidelity loss as a constraint to ensure the adversarial image from being deviated from the original image. We update the $\mathbf{x}^{\text{adv}}$ in latent space instead of in pixel space to ensure the naturalness of $\mathbf{x}^{\text{adv}}$.

rized image translations or editing, denoted as SDEdit$(\mathbf{x}, t)$, that manipulates the original input image $\mathbf{x}$. Our work aims to safeguard the input image $\mathbf{x}$ from the unauthorized manipulations by crafting an adversarial image $\mathbf{x}^{\text{adv}}$ through adding imperceptible perturbation to disrupt the reverse diffusion process of SDEdit for corrupted editions. For example, we want the main object of the image, e.g., the cat in the source image $\mathbf{x}$ as shown in Figure 2 is unable to be reconstructed by the reverse diffusion process. Meanwhile, the adversarial image should maintain similarity to the source image to ensure fidelity. The reason why we target SDEdit as our threat model is that it is recognized as the most common and general operation in diffusion-based unconditional image translation and conditional image editing. Additionally, it has been incorporated into various editing pipelines (Tsaban and Passos 2023; Zhang et al. 2023). Here we focus on the unconditional image translation for our main study, as they are essential in both unconditional and conditional editing pipelines. Formally, our objective to effectively safeguard images while maintaining fidelity is formulated as:

$$\max_{\mathbf{x}^{\text{adv}} \in \mathcal{M}} d(\text{SDEdit}(\mathbf{x}, t), \text{SDEdit}(\mathbf{x}^{\text{adv}}, t)) \quad \text{subject to } d'(\mathbf{x}, \mathbf{x}^{\text{adv}}) \leq \delta, \tag{1}$$

where $\mathcal{M}$ indicates natural image manifold, d and d' indicate image distance functions, and δ denotes the fidelity budget.

In the following sections, we first present a conceptual illustration of our method, followed by our framework for solving the optimization problem. We then discuss the novel design of our attacking loss and fidelity constraints, which provide more efficient criteria compared to previous methods. Finally, we introduce an advanced design to enhance adversarial image quality by latent optimization via victim-model-agnostic VAE.

3.2 Overview

To achieve effective protection against diffusion-based editing, we aim to push the adversarial image away from the original clean image by disrupting the intermediate step in the reverse diffusion process. For practical real-world applications, it's essential to ensure the adversarial image is perceptually similar to the original image. In practice, we uniformly sample the value of the forward diffusion step $t \sim [0, T]$ to generate noisy images and then perform optimization to craft the adversarial image $\mathbf{x}^{\text{adv}}$ via our attacking and fidelity losses, repeating the same process N times or until convergence. Figure 2 depicts these two push-and-pull criteria during different noise levels, the successful attack is represented in the light orange line where the reverse sample moves far away from the normal edition of the image. More specifically, our method can be formulated as follows:

$$\max_{\mathbf{x}^{\text{adv}} \in \mathcal{M}} \mathbb{E}_{t, \mathbf{x}_t | \mathbf{x}, \mathbf{x}_t^{\text{adv}} | \mathbf{x}} \mathcal{L}_{\text{attack}}(\mathbf{x}_t, \mathbf{x}_t^{\text{adv}}) \quad \text{subject to } \mathcal{L}_{\text{fidelity}}(\mathbf{x}, \mathbf{x}^{\text{adv}}) \leq \delta, \tag{2}$$

where δ denotes the attacking budget. The details of the attacking loss $\mathcal{L}_{\text{attack}}$ and the fidelity loss $\mathcal{L}_{\text{fidelity}}$ will be discussed in the following sections.

Framework. Our framework, shown in Figure 3, utilizes two identical and frozen victim UNets to extract feature representations from clean and adversarial images for our attacking loss calculation and a victim-model-agnostic VAE for the latent optimization strategy.

3.3 Proposed Losses

We propose two novel losses as optimization objectives to craft an adversarial example efficiently without running through all the diffusion steps. The attacking loss is designed to distract the feature representation of the denoising UNet; The fidelity loss is a constraint to ensure the adversarial image quality. For notation simplicity, we first define the samples $\mathbf{x}, \mathbf{x}^{\text{adv}}$ in different forwarded steps. Let $\mathcal{F}(\mathbf{x}, t, \epsilon) = \sqrt{\bar{\alpha}_t}\mathbf{x} + \sqrt{1-\bar{\alpha}_t}\epsilon$ be the diffusion forward process. Given timestep t sample from $[0, T]$, noises $\epsilon, \epsilon^{\text{adv}}$ sample from $\mathcal{N}(\mathbf{0}, \mathbf{I})$. We denote $\mathbf{x}_t = \mathcal{F}(\mathbf{x}, t, \epsilon)$, and $\mathbf{x}_t^{\text{adv}} = \mathcal{F}(\mathbf{x}^{\text{adv}}, t, \epsilon^{\text{adv}})$.

Attacking Loss. Our goal is to define effective criteria that could finally distract the reverse denoising process. PhotoGuard (Salman et al. 2023) proposed to backpropagate

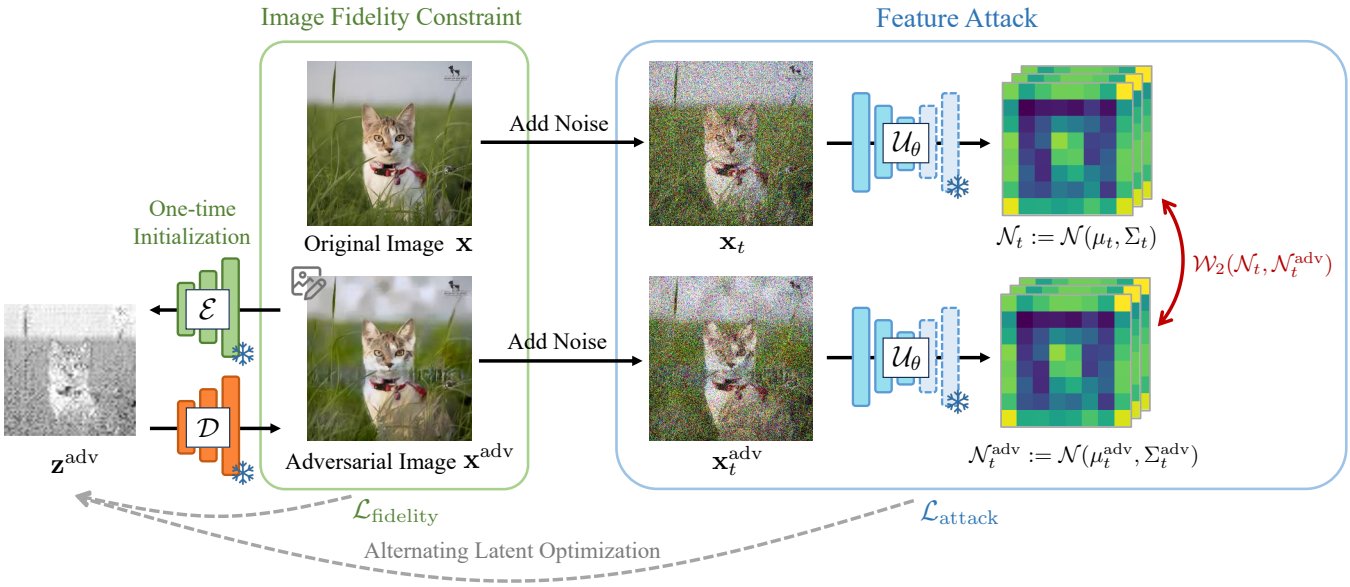

Figure 3: Overview of our AtkPDM$^+$ algorithm: Starting from the latent, $\mathbf{z}^{\text{adv}}$, of the initial adversarial image, we first decode back to pixel-domain to perform forward diffusion with both $\mathbf{x}$ and $\mathbf{x}^{\text{adv}}$ and feed them to frozen victim UNet. We then extract the feature representation of the middle block in UNet to calculate our $\mathcal{L}_{\text{attack}}$, aiming to distract the recognition of image semantics. We also calculate our $\mathcal{L}_{\text{fidelity}}$ in pixel-domain to constrain the optimization. Finally, the $\mathbf{z}^{\text{adv}}$ is being alternatively updated by loss gradients.

through all the steps of the reverse denoising process via PGD. However, this approach is prohibitively expensive, Diff-Protect (Xue et al. 2023) proposed to avoid the massive cost by leveraging Score Distillation (Poole et al. 2023) in optimization. Nevertheless, Diff-Protect relies heavily on gradients of attacking encoder of an LDM as stated in their results. In PDM, we don't have such an encoder to attack; however, we find that the denoising UNet has a similar structure to encoder-decoder models, and some previous works (Lin and Yang 2024; Li et al. 2023) characterize this property to accelerate and enhance the generation. From our observations of the feature roles in denoising UNets, we hypothesize that distracting specific inherent feature representation in UNet blocks could lead to effectively crafting an adversarial image. In practice, we first extract the feature representations of forwarded images $\mathbf{x}_t$ and $\mathbf{x}_t^{\text{adv}}$ in frozen UNet blocks of timestep t. Then, we adopt 2-Wasserstein distance (Arjovsky, Chintala, and Bottou 2017) to measure the discrepancy in the UNet feature space. The reason for choosing the 2-Wasserstein distance is that it better captures the distributional discrepancy via Optimal Transport Theory (Chen, Georgiou, and Tannenbaum 2018). Note that we aim to maximize the distance between $\mathbf{x}_t^{\text{adv}}$ and $\mathbf{x}_t$ in the UNet feature space to distract the denoising process. Formally, the attacking loss $\mathcal{L}_{\text{attack}}$ is defined as:

$$\mathcal{L}_{\text{attack}}(\mathbf{x}_t, \mathbf{x}_t^{\text{adv}}) = \mathcal{W}_2\left(\mathcal{U}_\theta^{(\text{mid})}(\mathbf{x}_t), \mathcal{U}_\theta^{(\text{mid})}(\mathbf{x}_t^{\text{adv}})\right). \quad (3)$$

Assuming the feature distributions approximate normal distributions expressed by mean μ_t and μ_t^{adv}, and non-singular covariance matrices Σ_t and Σ_t^{adv}. The calculation of the 2-Wasserstein distance between two normal distributions is viable through the closed-form solution (Dowson and Landau 1982; Olkin and Pukelsheim 1982; Chen, Georgiou, and Tannenbaum 2018):

$$\begin{aligned}\mathcal{W}_2^2(\mathcal{N}(\mu_t, \Sigma_t), \mathcal{N}(\mu_t^{\text{adv}}, \Sigma_t^{\text{adv}})) &= \|\mu_t - \mu_t^{\text{adv}}\|_2^2 \\ &+ \text{trace}(\Sigma_t + \Sigma_t^{\text{adv}} - 2(\Sigma_t^{\text{adv}\frac{1}{2}}\Sigma_t\Sigma_t^{\text{adv}\frac{1}{2}})^{\frac{1}{2}}).\end{aligned} \quad (4)$$

Fidelity Loss. To control the attack budget for adversarial image quality, we design a constraint function that utilizes the feature extractor from a pretrained classifier to calculate the fidelity loss. In our case, we sum up the 2-Wasserstein feature losses of L different layers. Specifically, we define $\mathcal{L}_{\text{fidelity}}$ as:

$$\mathcal{L}_{\text{fidelity}}(\mathbf{x}_t, \mathbf{x}_t^{\text{adv}}) = \sum_{\ell=1}^{L} \mathcal{W}_2(\phi_\ell(\mathbf{x}), \phi_\ell(\mathbf{x}^{\text{adv}})), \quad (5)$$

where $\mathcal{W}_2$ denotes 2-Wasserstein distance and ϕ_ℓ denotes layer ℓ of the feature extractor.

3.4 Alternating Optimization for Adversarial Image

We solve the constrained optimization problem via alternating optimization to craft the adversarial images, detailed optimization loop of AtkPDM$^+$ is provided in Algorithm 1. To maximize the $\mathcal{L}_{\text{attack}}$, we take the negative $\mathcal{L}_{\text{attack}}$ and perform gradient descent. AtkPDM algorithm and the derivation of the alternating optimization are provided in Appendix.

Algorithm 1: AtkPDM$^+$

Input: Image to be protected $\mathbf{x}$, attack budget $\delta > 0$, step size $\gamma_{\text{attack}}, \gamma_{\text{fidelity}} > 0$, VAE encoder $\mathcal{E}$, and VAE decoder $\mathcal{D}$
Initialization: $\mathbf{x}^{\text{adv}} \leftarrow \mathbf{x}$, $L_{\text{attack}} \leftarrow \infty$
Encode adversarial image to latent space: $\mathbf{z}^{\text{adv}} \leftarrow \mathcal{E}(\mathbf{x}^{\text{adv}})$
while L_{attack} not convergent **do**
    Decode adversarial latent to pixel space: $\mathbf{x}^{\text{adv}} \leftarrow \mathcal{D}(\mathbf{z}^{\text{adv}})$
    Sample timestep: $t \sim [0, T]$
    Sample noise: $\epsilon, \epsilon^{\text{adv}} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
    Compute original noisy sample: $\mathbf{x}_t \leftarrow \mathcal{F}(\mathbf{x}, t, \epsilon)$
    Compute adversarial noisy sample: $\mathbf{x}_t^{\text{adv}} \leftarrow \mathcal{F}(\mathbf{x}^{\text{adv}}, t, \epsilon^{\text{adv}})$
    Update $\mathbf{z}^{\text{adv}}$ by Gradient Descent: $\mathbf{z}^{\text{adv}} \leftarrow \mathbf{z}^{\text{adv}} - \gamma_{\text{attack}} \operatorname{sign}(\nabla_{\mathbf{z}^{\text{adv}}}(-\mathcal{L}_{\text{attack}}(\mathbf{x}_t, \mathbf{x}_t^{\text{adv}})))$
    while $\mathcal{L}_{\text{fidelity}}(\mathbf{x}, \mathcal{D}(\mathbf{z}^{\text{adv}})) > \delta$ **do**
        $\mathbf{z}^{\text{adv}} \leftarrow \mathbf{z}^{\text{adv}} - \gamma_{\text{fidelity}} \nabla_{\mathbf{z}^{\text{adv}}} \mathcal{L}_{\text{fidelity}}(\mathbf{x}, \mathcal{D}(\mathbf{z}^{\text{adv}}))$
    end while
end while
Decode adversarial latent to pixel space: $\mathbf{x}^{\text{adv}} \leftarrow \mathcal{D}(\mathbf{z}^{\text{adv}})$
return $\mathbf{x}^{\text{adv}}$

3.5 Latent Optimization via Pretrained-VAE

Previous works suggest that diffusion models have a strong capability against adversarial perturbations (Xue and Chen 2024), making them hard to be attacked via pixel-domain optimization. Moreover, they are even considered as good purifiers of adversarial perturbations (Nie et al. 2022).

Here, we propose a strategy that crafts the perturbation in the latent space of the pre-trained Variational Autoencoder (VAE) (Kingma and Welling 2014), and the gradients are used to update the latent. After N iterations or losses converge, we decode back via the decoder $\mathcal{D}$ to pixel domain as our final adversarial image. The motivation for adopting VAE is inspired by MPGD (He et al. 2024). This strategy is effective for crafting a robust adversarial image against pixel-domain diffusion models while also better preserving the adversarial image quality rather than only incorporating fidelity constraints. Note that, ideally, manifold preservation is guaranteed when using perfect VAE. In practice, we use the best available LDM's VAE agnostic to the victim model as our latent optimization VAE. Detailed latent optimization loop is provided in Algorithm 1.

4 Experiment Results

4.1 Experiment Settings

Implementation Details. We conduct all our experiments in white box settings and examine the effectiveness of our attacks using SDEdit (Meng et al. 2021). For the VAE (Kingma and Welling 2014) in our AtkPDM$^+$, we utilize the one provided by StableDiffusion v1.5 (Rombach et al. 2022). We run all of our experiments with 300 optimization steps, which empirically determined, balancing attacking effectiveness and adversarial image quality with a reasonable speed. Other loss parameters and running time are provided in the Appendix. The implementation is built on the Diffusers library (von Platen et al. 2022). All the experiments are conducted with a single Nvidia Tesla V100 GPU.

Victim Models and Datasets. We test our approach on PDMs with three open-source checkpoints on HuggingFace, specifically "google/ddpm-ema-church-256", "google/ddpm-cat-256" and "google/ddpm-ema-celebahq-256". For the results reported in Table 1, we run 30 images for each victim model. Additionally, for generalizability in practical scenarios, we synthesize the data with half randomly selected from the originally trained dataset and another half from randomly crawled with keywords from the Internet.

Baseline Methods and Evaluation Metrics. To the best of our knowledge, the previous methods have mainly focused on LDMs, and effective PDM attacks have not yet been developed, however, we still implement AdvDM (Liang et al. 2023) with the proposed semantic loss by (Salman et al. 2023; Liang et al. 2023; Liang and Wu 2023; Xue et al. 2023) for comparison. Notably, Diff-Protect (Xue et al. 2023) proposed to minimize the semantic loss and is counterintuitively better than maximizing the semantic loss. We also adopt this method in attacking PDMs. To quantify the adversarial image visual quality, we adopt Structural Similarity (SSIM) (Wang et al. 2004), Peak Signal-to-Noise Ratio (PSNR), and Learned Perceptual Image Patch Similarity (LPIPS) (Zhang et al. 2018) as the evaluation metrics but negatively quantify the effectiveness of our attack. We also adopt the Image Alignment Score (IA-Score) (Kumari et al. 2023) that leverages CLIP (Radford et al. 2021) to calculate the cosine similarity between two image encoder features. In distinguishing from the previous methods, to more faithfully reflect the attacking effectiveness, we fix the same seed of the random generator when generating clean and adversarial samples, then calculating the scores based on the paired samples.

4.2 Attacking Effectiveness on PDMs

As quantitatively reported in Table 1 and qualitative results in Figure 4, compared to the previous PGD-based methods incorporating semantic loss, i.e., negative training loss of diffusion models, our method exhibits superior performance in both adversarial image quality and attacking effectiveness. In addition, our reported numbers are generally stable, as reflected in lower standard deviation. It is worth noting that even if the adversarial image qualities of the PGD-based methods are far worse than ours, their attacking effectiveness still falls short, suggesting that PDMs are robust against traditional perturbation methods. This finding is also aligned with previous works (Xue et al. 2023; Xue and Chen 2024). For AtkPDM$^+$, combined with our latent optimization strategy, the adversarial image quality has been enhanced while slightly affecting the attacking effectiveness, still outperforming the previous methods. Besides unconditional PDMs, we also compare with the previous best method Diff-Protect against a conditional PDM DeepFloyd IF (at StabilityAI 2023), reported in Table 2. Although the attacking effectiveness of Diff-Protect seems better than ours, this may be due to their adversarial image

	Methods	Adversarial Image Quality			Attacking Effectiveness			
		SSIM ↑	PSNR ↑	LPIPS ↓	SSIM ↓	PSNR ↓	LPIPS ↑	IA-Score ↓
Church	AdvDM (Liang et al. 2023)	0.37 ± 0.09	28.17 ± 0.22	0.73 ± 0.16	0.89 ± 0.05	31.06 ± 1.94	0.17 ± 0.09	0.93 ± 0.04
	Diff-Protect (Xue et al. 2023)	0.39 ± 0.07	28.03 ± 0.12	0.67 ± 0.11	0.82 ± 0.05	31.90 ± 1.08	0.23 ± 0.07	0.91 ± 0.04
	AtkPDM (Ours)	0.75 ± 0.03	28.22 ± 0.10	0.26 ± 0.04	**0.75** ± 0.04	**29.61** ± 0.23	**0.40** ± 0.05	**0.76** ± 0.06
	AtkPDM$^+$ (Ours)	**0.81** ± 0.03	**28.64** ± 0.19	**0.13** ± 0.02	0.79 ± 0.04	30.05 ± 0.47	0.33 ± 0.07	0.81 ± 0.06
Cat	AdvDM (Liang et al. 2023)	0.48 ± 0.09	28.34 ± 0.18	0.65 ± 0.12	0.96 ± 0.02	32.32 ± 2.49	0.10 ± 0.05	0.97 ± 0.03
	Diff-Protect (Xue et al. 2023)	0.33 ± 0.10	28.03 ± 0.15	0.80 ± 0.15	0.90 ± 0.05	33.94 ± 1.93	0.18 ± 0.08	0.95 ± 0.03
	AtkPDM (Ours)	0.71 ± 0.06	28.47 ± 0.18	0.29 ± 0.05	**0.83** ± 0.03	**30.73** ± 0.51	**0.39** ± 0.05	**0.81** ± 0.04
	AtkPDM$^+$ (Ours)	**0.83** ± 0.04	**29.41** ± 0.37	**0.09** ± 0.02	0.93 ± 0.01	33.02 ± 0.74	0.18 ± 0.02	0.92 ± 0.01
Face	AdvDM (Liang et al. 2023)	0.48 ± 0.05	**28.75** ± 0.18	0.64 ± 0.10	0.99 ± 0.00	37.96 ± 1.75	0.02 ± 0.01	0.99 ± 0.00
	Diff-Protect (Xue et al. 2023)	0.25 ± 0.04	28.09 ± 0.20	0.91 ± 0.11	0.95 ± 0.02	35.33 ± 1.62	0.08 ± 0.04	0.96 ± 0.02
	AtkPDM (Ours)	0.56 ± 0.04	28.01 ± 0.22	0.36 ± 0.04	**0.74** ± 0.03	**29.14** ± 0.36	**0.40** ± 0.05	**0.62** ± 0.07
	AtkPDM$^+$ (Ours)	**0.81** ± 0.04	28.39 ± 0.20	**0.12** ± 0.03	0.86 ± 0.03	30.26 ± 0.72	0.24 ± 0.07	0.80 ± 0.08

Table 1: Quantitative results in attacking different unconditional PDMs. The best is marked in bold and the second best is underlined. Errors denote one standard deviation of all images in our test datasets.

Methods	Adversarial Image Quality			Attacking Effectiveness			
	SSIM ↑	PSNR ↑	LPIPS ↓	SSIM ↓	PSNR ↓	LPIPS ↑	IA-Score ↓
Diff-Protect (Xue et al. 2023)	0.47 ± 0.08	27.96 ± 0.08	0.46 ± 0.05	**0.49** ± 0.10	**28.13** ± 0.15	**0.36** ± 0.10	**0.79** ± 0.06
AtkPDM$^+$ (Ours)	**0.79** ± 0.06	**28.48** ± 0.33	**0.06** ± 0.02	0.72 ± 0.10	28.50 ± 0.48	0.10 ± 0.04	0.86 ± 0.08

Table 2: Quantitative results in attacking conditional PDM DeepFloyd IF. The best is marked in bold and the second best is underlined. Errors denote one standard deviation of all images in our test datasets.

Defense Method	Attacking Effectiveness			
	SSIM ↓	PSNR ↓	LPIPS ↑	IA-Score ↓
LDM-Pure	0.78	29.84	0.35	0.80
Crop-and-Resize	0.68	29.28	0.42	0.79
JPEG Comp.	0.78	29.82	0.36	0.79
None	0.79	30.05	0.33	0.81

Table 3: Quantitative results of our adversarial images against defense methods. LDM-Pure, Crop-and-Resize, and JPEG Compression fail to defend our attack. "None" indicates no defense is applied, as the baseline for comparison.

Setting	Attacking Effectiveness			
	SSIM ↓	PSNR ↓	LPIPS ↑	IA-Score ↓
White Box	0.79	30.05	0.33	0.81
Black Box	0.86	30.25	0.29	0.85
Difference	0.07	0.20	0.04	0.04

Table 4: Quantitative results of black box attack. We use the same set of adversarial images and feed to white box and black box models to examine the black box transferability.

quality being severely corrupted during the attack. Hence, it cannot fulfill our two objectives simultaneously. In addition, our framework is extensible to attack LDMs, please refer to Appendix provided in the project page.

4.3 Black Box Transferability

We craft adversarial images with the proxy model, "google/ddpm-ema-church-256", in white-box settings and test their transferability against "google/ddpm-bedroom-256" model as black-box attacks. Under identical validation settings, Table 4 reveals only a slight decrease in attack effectiveness metrics, suggesting black-box transferability.

4.4 Robustness Against Defense Methods

We examine the robustness of our approach against three widely recognized and effective adversarial defense methods. The quantitative results in Table 3 demonstrate that our method is robust against these three defense methods, with four metrics listed in Table 3 not worse than no defenses. Surprisingly, these defense methods even make the adversarial image more effective than cases without defense. We provide the implementation details of each defense method in the following sections.

LDM Purification. Nie at. al. proposed DiffPure (Nie et al. 2022) that leverages a pre-trained Diffusion Model to purify adversarial images targeting classifier models to defend effectively. The purification process is essentially an unconditional SDEdit process with small forward t. Here, we use a pre-trained LDM (StableDiffusion v1.5) and $t = 100$ to purify our adversarial image as a defense method.

Crop and Resize. Noted by Diff-Protect, "crop and resize" is a simple yet the most effective defense method against their attacks on LDMs. We test our method against this defense using their settings, i.e., cropping 20% of the adversarial image and resizing it to its original dimensions.

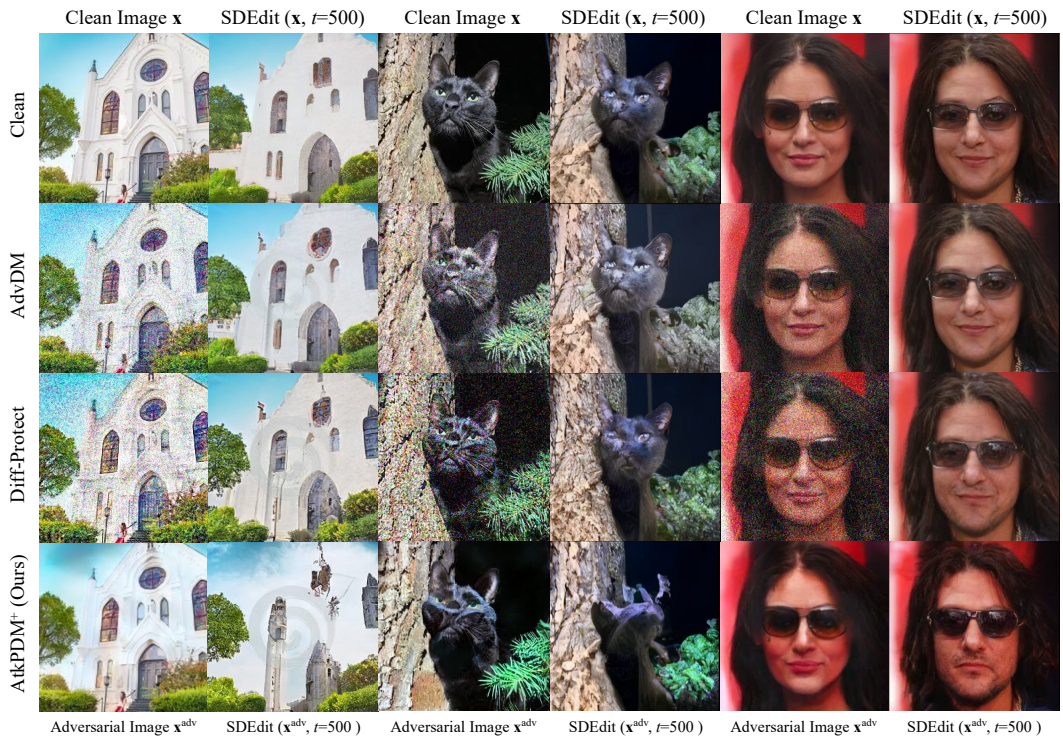

Figure 4: Qualitative results compared to the previous methods. Our adversarial images can effectively corrupt the edited results without significant fidelity decrease. The same column shares the same random seed for fair comparisons.

Losses	VAE	Adversarial Image Quality			Attacking Effectiveness			
		SSIM ↑	PSNR ↑	LPIPS ↓	SSIM ↓	PSNR ↓	LPIPS ↑	IA-Score ↓
$\mathcal{L}_{\text{semantic}}$		0.37 ± 0.09	28.17 ± 0.22	0.73 ± 0.16	0.89 ± 0.05	31.06 ± 1.94	0.17 ± 0.09	0.93 ± 0.04
$\mathcal{L}_{\text{semantic}}$	✓	0.80 ± 0.05	<u>29.78</u> ± 0.42	<u>0.17</u> ± 0.03	0.82 ± 0.05	30.43 ± 0.75	0.15 ± 0.06	0.92 ± 0.04
$\mathcal{L}_{\text{semantic}} + \mathcal{L}_{\text{fidelity}}$	✓	**0.82** ± 0.05	**30.30** ± 0.81	**0.13** ± 0.03	0.90 ± 0.03	31.24 ± 1.19	0.08 ± 0.03	0.96 ± 0.02
$\mathcal{L}_{\text{attack}} + \mathcal{L}_{\text{fidelity}}$ (AtkPDM)		0.75 ± 0.03	28.22 ± 0.10	0.26 ± 0.04	**0.75** ± 0.04	**29.61** ± 0.23	**0.40** ± 0.05	**0.76** ± 0.06
$\mathcal{L}_{\text{attack}} + \mathcal{L}_{\text{fidelity}}$ (AtkPDM$^+$)	✓	<u>0.81</u> ± 0.03	28.64 ± 0.19	**0.13** ± 0.02	<u>0.79</u> ± 0.04	<u>30.05</u> ± 0.47	<u>0.33</u> ± 0.07	<u>0.81</u> ± 0.06

Table 5: Quantitative results of ablation study. The best is marked in bold and the second best is underlined. Errors denote one standard deviation of all images in our test datasets.

JPEG Compression. Sandoval-Segura et al. (Sandoval-Segura, Geiping, and Goldstein 2023) demonstrated that JPEG compression is a simple yet effective adversarial defense method. In our experiments, we implement the JPEG compression at a quality setting of 25%.

4.5 Effectiveness of Latent Optimization via VAE

We first incorporate our VAE latent optimization strategy in the previous semantic-loss-based methods. From Table 5, without using $\mathcal{L}_{\text{fidelity}}$, latent optimization has significantly enhanced the adversarial image quality and even slightly improved the attacking effectiveness. Adopting latent optimization in our approach enhances visual quality with a negligible decrease in attacking effectiveness. Surprisingly, incorporating our $\mathcal{L}_{\text{fidelity}}$ with current PGD-based method will drastically decrease the adversarial image quality despite its attack performing better than ours. This may be due to different constrained optimization problem settings.

5 Conclusion

This paper presents the first framework to protect against image manipulation by Pixel-domain Diffusion Models (PDMs). While denoising UNets withstand traditional PGD attacks, their feature space remains vulnerable. Our feature attacking loss exploits these vulnerabilities, generating adversarial images that mislead PDMs, resulting in corrupted output. We approach this image protection problem as a constrained optimization problem, solving it through alternating optimization. Furthermore, our latent optimization strategy via VAE enhances the naturalness of our adversarial images. Extensive experiments validate the efficacy of our method, achieving state-of-the-art performance in attacking PDMs.

Acknowledgements

This research is supported by National Science and Technology Council, Taiwan (R.O.C) under the grant numbers NSTC-113-2634-F-002-007, NSTC-112-2222-E-001-001-MY2, NSTC-113-2634-F-001-002-MBK, NSTC-113-2221-E-002-201, and Academia Sinica under the grant number of AS-CDA-110-M09. We thank to National Center for High-performance Computing (NCHC) of National Applied Research Laboratories (NARLabs) in Taiwan for providing computational and storage resources.

References

Arjovsky, M.; Chintala, S.; and Bottou, L. 2017. Wasserstein generative adversarial networks. In *International Conference on Machine Learning (ICML)*.

at StabilityAI, D. L. 2023. DeepFloyd IF: a novel state-of-the-art open-source text-to-image model with a high degree of photorealism and language understanding. https://www.deepfloyd.ai/deepfloyd-if.

Chen, Y.; Georgiou, T. T.; and Tannenbaum, A. 2018. Optimal transport for Gaussian mixture models. *IEEE Access*.

Dhariwal, P.; and Nichol, A. 2021. Diffusion models beat gans on image synthesis. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Dowson, D.; and Landau, B. 1982. The Fréchet distance between multivariate normal distributions. *Journal of multivariate analysis*.

Efron, B. 2011. Tweedie's formula and selection bias. *Journal of the American Statistical Association*.

Goodfellow, I.; Shlens, J.; and Szegedy, C. 2015. Explaining and Harnessing Adversarial Examples. In *International Conference on Learning Representations (ICLR)*.

He, Y.; Murata, N.; Lai, C.-H.; Takida, Y.; Uesaka, T.; Kim, D.; Liao, W.-H.; Mitsufuji, Y.; Kolter, J. Z.; Salakhutdinov, R.; and Ermon, S. 2024. Manifold Preserving Guided Diffusion. In *International Conference on Learning Representations (ICLR)*.

Hertz, A.; Mokady, R.; Tenenbaum, J.; Aberman, K.; Pritch, Y.; and Cohen-or, D. 2023. Prompt-to-Prompt Image Editing with Cross-Attention Control. In *International Conference on Learning Representations (ICLR)*.

Ho, J.; Jain, A.; and Abbeel, P. 2020. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Kingma, D. P.; and Welling, M. 2014. Auto-Encoding Variational Bayes. In *International Conference on Learning Representations (ICLR)*.

Kumari, N.; Zhang, B.; Zhang, R.; Shechtman, E.; and Zhu, J.-Y. 2023. Multi-concept customization of text-to-image diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Laidlaw, C.; Singla, S.; and Feizi, S. 2021. Perceptual Adversarial Robustness: Defense Against Unseen Threat Models. In *International Conference on Learning Representations (ICLR)*.

Li, S.; Hu, T.; Khan, F. S.; Li, L.; Yang, S.; Wang, Y.; Cheng, M.-M.; and Yang, J. 2023. Faster Diffusion: Rethinking the Role of UNet Encoder in Diffusion Models. *arXiv preprint arXiv:2312.09608*.

Liang, C.; and Wu, X. 2023. Mist: Towards Improved Adversarial Examples for Diffusion Models. *arXiv preprint arXiv:2305.12683*.

Liang, C.; Wu, X.; Hua, Y.; Zhang, J.; Xue, Y.; Song, T.; Xue, Z.; Ma, R.; and Guan, H. 2023. Adversarial Example Does Good: Preventing Painting Imitation from Diffusion Models via Adversarial Examples. In *International Conference on Machine Learning (ICML)*.

Lin, S.; and Yang, X. 2024. Diffusion Model with Perceptual Loss. *arXiv preprint arXiv:2401.00110*.

Liu, J.; Wei, C.; Guo, Y.; Yu, H.; Yuille, A.; Feizi, S.; Lau, C. P.; and Chellappa, R. 2023. Instruct2Attack: Language-Guided Semantic Adversarial Attacks. *arXiv preprint arXiv:2311.15551*.

Lo, L.; Yeo, C. Y.; Shuai, H.-H.; and Cheng, W.-H. 2024. Distraction is All You Need: Memory-Efficient Image Immunization against Diffusion-Based Image Editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Madry, A.; Makelov, A.; Schmidt, L.; Tsipras, D.; and Vladu, A. 2018. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations (ICLR)*.

Meng, C.; He, Y.; Song, Y.; Song, J.; Wu, J.; Zhu, J.-Y.; and Ermon, S. 2021. SDEdit: Guided Image Synthesis and Editing with Stochastic Differential Equations. In *International Conference on Learning Representations (ICLR)*.

Mokady, R.; Hertz, A.; Aberman, K.; Pritch, Y.; and Cohen-Or, D. 2023. NULL-Text Inversion for Editing Real Images Using Guided Diffusion Models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Nie, W.; Guo, B.; Huang, Y.; Xiao, C.; Vahdat, A.; and Anandkumar, A. 2022. Diffusion Models for Adversarial Purification. In *International Conference on Machine Learning (ICML)*.

Olkin, I.; and Pukelsheim, F. 1982. The distance between two random vectors with given dispersion matrices. *Linear Algebra and its Applications*.

Parmar, G.; Kumar Singh, K.; Zhang, R.; Li, Y.; Lu, J.; and Zhu, J.-Y. 2023. Zero-shot image-to-image translation. In *ACM SIGGRAPH 2023 Conference Proceedings*.

Poole, B.; Jain, A.; Barron, J. T.; and Mildenhall, B. 2023. DreamFusion: Text-to-3D using 2D Diffusion. In *International Conference on Learning Representations (ICLR)*.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. 2021. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning (ICML)*.

Rombach, R.; Blattmann, A.; Lorenz, D.; Esser, P.; and Ommer, B. 2022. High-resolution image synthesis with latent

diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Salman, H.; Khaddaj, A.; Leclerc, G.; Ilyas, A.; and Madry, A. 2023. Raising the cost of malicious AI-powered image editing. In *International Conference on Machine Learning (ICML)*.

Sandoval-Segura, P.; Geiping, J.; and Goldstein, T. 2023. JPEG compressed images can bypass protections against ai editing. *arXiv preprint arXiv:2304.02234*.

Shan, S.; Cryan, J.; Wenger, E.; Zheng, H.; Hanocka, R.; and Zhao, B. Y. 2023. Glaze: Protecting artists from style mimicry by Text-to-Image models. In *USENIX Security Symposium*.

Shan, S.; Ding, W.; Passananti, J.; Wu, S.; Zheng, H.; and Zhao, B. Y. 2024. Nightshade: Prompt-Specific Poisoning Attacks on Text-to-Image Generative Models. In *2024 IEEE Symposium on Security and Privacy (SP)*.

Simonyan, K.; and Zisserman, A. 2014. Very Deep Convolutional Networks for Large-Scale Image Recognition. *CoRR*.

Song, J.; Meng, C.; and Ermon, S. 2021. Denoising diffusion implicit models. In *International Conference on Learning Representations (ICLR)*.

Song, Y.; Garg, S.; Shi, J.; and Ermon, S. 2020. Sliced score matching: A scalable approach to density and score estimation. In *Uncertainty in Artificial Intelligence*.

Tsaban, L.; and Passos, A. 2023. LEDITS: Real Image Editing with DDPM Inversion and Semantic Guidance. *arXiv preprint arXiv:2307.00522*.

Tumanyan, N.; Geyer, M.; Bagon, S.; and Dekel, T. 2023. Plug-and-play diffusion features for text-driven image-to-image translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

von Platen, P.; Patil, S.; Lozhkov, A.; Cuenca, P.; Lambert, N.; Rasul, K.; Davaadorj, M.; Nair, D.; Paul, S.; Berman, W.; Xu, Y.; Liu, S.; and Wolf, T. 2022. Diffusers: State-of-the-art diffusion models. https://github.com/huggingface/diffusers.

Wang, Z.; Bovik, A. C.; Sheikh, H. R.; and Simoncelli, E. P. 2004. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing (TIP)*.

Wang, Z.; Zhao, L.; and Xing, W. 2023. Stylediffusion: Controllable disentangled style transfer via diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

Xue, H.; Araujo, A.; Hu, B.; and Chen, Y. 2024. Diffusion-based adversarial sample generation for improved stealthiness and controllability. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Xue, H.; and Chen, Y. 2024. Pixel is a Barrier: Diffusion Models Are More Adversarially Robust Than We Think. *arXiv preprint arXiv:2404.13320*.

Xue, H.; Liang, C.; Wu, X.; and Chen, Y. 2023. Toward effective protection against diffusion-based mimicry through score distillation. In *International Conference on Learning Representations (ICLR)*.

Zhang, R.; Isola, P.; Efros, A. A.; Shechtman, E.; and Wang, O. 2018. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Zhang, Y.; Huang, N.; Tang, F.; Huang, H.; Ma, C.; Dong, W.; and Xu, C. 2023. Inversion-based style transfer with diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Supplementary Material

A More Implementation Details

The feature extractor for calculating $\mathcal{L}_{\text{fidelity}}$ is VGG16 (Simonyan and Zisserman 2014) with IMAGENET1K-V1 checkpoint. We use the SDEdit with the forward step $t = 500$ for our main study results as it balances faithfulness to the original image and flexibility for editing. Empirically, we choose to randomly sample the forward step $t \sim [0, 500]$ to enhance the optimization efficiency. The average time to optimize 300 steps for an image on a single Nvidia Tesla V100 is about 300 seconds. The estimated average memory usage is about 24GB. Table 6 provides the details of the step sizes that we use to attack different models.

Models	Step Size	
	γ_{attack}	γ_{fidelity}
google/ddpm-ema-church-256	$100/255$	$40/255$
google/ddpm-cat-256	$100/255$	$5/255$
google/ddpm-ema-celebahq-256	$50/255$	$35/255$

Table 6: The step sizes used for different models during optimization.

B More Experimental Results

B.1 Attack Effectiveness on Latent Diffusion Models

We propose the feature representation attacking loss which can be adapted to target any UNet-based diffusion models. Hence, it is applicable to attack LDM using our proposed framework. We follow the evaluation settings of the previous work (Xue et al. 2023) for fair comparisons. Quantitative results are shown in Table 7. Compared to previous LDM-specified methods (Liang et al. 2023; Liang and Wu 2023; Xue et al. 2023), our method could achieve comparable results. This finding reflects the general vulnerability in UNet-based diffusion models that can be exploited to craft adversarial images against either PDMs or LDMs.

B.2 Qualitative Demonstration of Corrupting UNet Feature during Sampling

We qualitatively show an example of our attack effectiveness regarding UNet representation discrepancies in Figure 6. We compare a clean and an adversarial image using the same denoising process. Then, we take the feature maps of the second-last decoder block layer, close to the final predicted noise, to demonstrate their recognition of input image semantics. The results in Figure 6 show that from t = 500, the feature maps of each pair start with a similar structure, then as the t decreases, the feature maps gradually have higher discrepancies, suggesting our method, by attacking the middle representation of UNet, can effectively disrupt the reverse denoising process and mislead to corrupted samples.

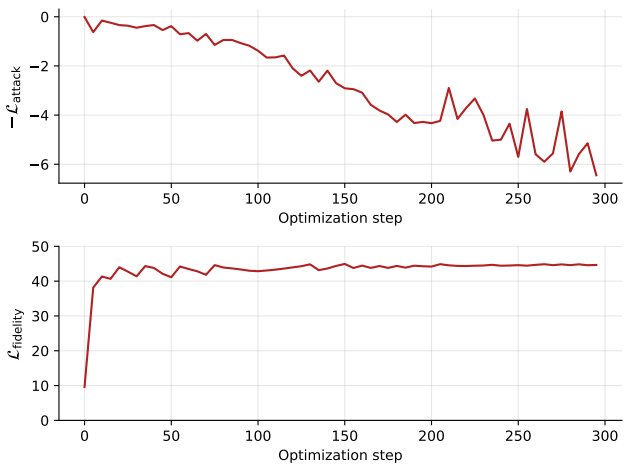

Figure 5: Loss curves of our $\mathcal{L}_{\text{attack}}$ and $\mathcal{L}_{\text{fidelity}}$ against optimization step.

B.3 Qualitative Results of Loss Ablation

Figure 7 presents qualitative results of loss ablation where i., ii., and iii. indicate performing PGAscent with different configurations. i. utilizes only semantic loss; ii. utilizes semantic loss with our latent optimization strategy; iii. utilizes both semantic loss, our proposed $\mathcal{L}_{\text{fidelity}}$ and latent optimization. The results show that our $\mathcal{L}_{\text{fidelity}}$ and latent optimization can enhance the adversarial image quality of PGAscent. Moreover, comparing our proposed two methods, AtkPDM$^+$ generates a more natural adversarial image than AtkPDM while maintaining attack effectiveness.

B.4 Example of Loss Curves

Figure 5 shows an example of our loss trends among optimization steps. $\mathcal{L}_{\text{attack}}$ has decreasing trend as the optimization step increases. $\mathcal{L}_{\text{fidelity}}$ has an increasing trend and converges to satisfy the constraint of the attack budget δ.

B.5 Different Forward Time-step Sampling

When using Monte Carlo sampling for optimization, the forward time step t^* is sampled uniformly. We study the scenario that when t^* is fixed for optimization. As shown in Figure 8, a primary result shows that when attacking $t^* = 400$ to $t^* = 500$, the attacking effectiveness is better than other time steps. In practice, we can not know user-specified t^* for editing in advance; however, this suggests that diffusion models have a potential temporal vulnerability that can be further exploited to increase efficiency.

B.6 More Qualitative Results

We provide more qualitative results in Figure 9 to showcase that our method can significantly change or corrupt the generated results with little modification on adversarial images. In contrast, previous methods add obvious perturbation to adversarial images but still fail to change the edited results to achieve the safeguarding goal.

	Methods	Adversarial Image Quality			Attacking Effectiveness			
		SSIM ↑	PSNR ↑	LPIPS ↓	SSIM ↓	PSNR ↓	LPIPS ↑	IA-Score ↓
Church	AdvDM (Liang et al. 2023)	**0.85** ± 0.03	**30.42** ± 0.15	0.23 ± 0.06	0.19 ± 0.05	28.00 ± 0.16	0.71 ± 0.04	0.49 ± 0.06
	Mist (Liang and Wu 2023)	0.81 ± 0.03	29.45 ± 0.13	0.25 ± 0.05	**0.14** ± 0.03	**27.95** ± 0.13	**0.76** ± 0.04	0.48 ± 0.05
	Diff-Protect (Xue et al. 2023)	0.79 ± 0.03	29.92 ± 0.15	0.24 ± 0.06	0.15 ± 0.03	28.00 ± 0.14	0.71 ± 0.04	0.48 ± 0.05
	AtkPDM (Ours)	0.82 ± 0.02	30.40 ± 0.27	0.24 ± 0.05	**0.14** ± 0.03	27.96 ± 0.17	0.74 ± 0.02	**0.47** ± 0.04
	AtkPDM$^+$ (Ours)	0.61 ± 0.07	29.17 ± 0.32	**0.20** ± 0.02	0.27 ± 0.06	28.07 ± 0.18	0.66 ± 0.05	0.51 ± 0.06
Cat	AdvDM (Liang et al. 2023)	**0.86** ± 0.04	30.68 ± 0.24	0.25 ± 0.09	0.21 ± 0.05	28.03 ± 0.21	0.70 ± 0.07	0.53 ± 0.04
	Mist (Liang and Wu 2023)	0.81 ± 0.04	29.63 ± 0.22	0.27 ± 0.08	**0.14** ± 0.04	**27.96** ± 0.17	**0.77** ± 0.06	**0.52** ± 0.04
	Diff-Protect (Xue et al. 2023)	0.78 ± 0.05	30.12 ± 0.24	0.27 ± 0.08	0.16 ± 0.05	**27.96** ± 0.15	0.72 ± 0.06	**0.52** ± 0.03
	AtkPDM (Ours)	0.84 ± 0.02	**30.79** ± 0.49	0.25 ± 0.07	0.18 ± 0.04	28.00 ± 0.19	0.72 ± 0.05	**0.52** ± 0.03
	AtkPDM$^+$ (Ours)	0.68 ± 0.13	29.68 ± 0.74	**0.16** ± 0.03	0.31 ± 0.10	28.13 ± 0.27	0.64 ± 0.06	0.54 ± 0.04
Face	AdvDM (Liang et al. 2023)	**0.83** ± 0.02	30.81 ± 0.22	0.32 ± 0.06	0.26 ± 0.05	28.07 ± 0.28	0.74 ± 0.05	0.47 ± 0.07
	Mist (Liang and Wu 2023)	0.79 ± 0.03	29.75 ± 0.22	0.34 ± 0.06	**0.19** ± 0.05	**27.99** ± 0.21	**0.81** ± 0.05	0.46 ± 0.08
	Diff-Protect (Xue et al. 2023)	0.74 ± 0.04	30.34 ± 0.13	0.33 ± 0.06	0.21 ± 0.05	28.03 ± 0.21	0.76 ± 0.06	0.45 ± 0.07
	AtkPDM (Ours)	**0.83** ± 0.02	**31.21** ± 0.44	0.31 ± 0.05	0.21 ± 0.04	28.03 ± 0.26	0.78 ± 0.04	**0.44** ± 0.06
	AtkPDM$^+$ (Ours)	0.82 ± 0.05	30.05 ± 0.51	**0.14** ± 0.03	0.41 ± 0.08	28.24 ± 0.39	0.63 ± 0.07	0.52 ± 0.07

Table 7: Quantitative results in attacking LDM. The best is marked in bold and the second best is underlined. Errors denote one standard deviation of all images in our test datasets.

C Backgrounds of Diffusion Models

Score-based models and diffusion models allowing generate samples starting from easy-to-sample Gaussian noise to complex target distributions. Starting from Gaussian noise, the sampling process iteratively applies the score function, i.e., $\nabla_{\mathbf{x}} \log p(\mathbf{x})$ of the complex target distribution $p(\mathbf{x})$ to generate the sample from $p(\mathbf{x})$. The exact estimation of the ground truth score function is intractable since the score function is the derivative of the target distribution $p(\mathbf{x})$. However, we can approximate the score function without directly dealing with $p(\mathbf{x})$. Song et al. proposed score-based models (Song et al. 2020) to learn the score function effectively via score matching. Ho et al. proposed Denoising Diffusion Probability Model (DDPM) (Ho, Jain, and Abbeel 2020), providing another perspective on learning score function with noise perturbed data, allowing more effective low-density area estimation and improving the mode diversity, thereby capable of generating highly sophisticated data, e.g., natural images. In a nutshell, training DDPM involves perturbing data with Gaussian noise in different timestep-controlled variance schedules, i.e., forward diffusion, and a parametrized model $\epsilon_\theta(\mathbf{x}_t, t)$ will learn to predict the added noise conditioning on noisy data $\mathbf{x}_t$ and current noise level t. Sampling with learned DDPM starts with random noise and iteratively applies the model $\epsilon_\theta(\mathbf{x}_t, t)$ to denoise, i.e., reverse diffusion sampling, thereby generating a sample from the learned distribution. Specifically, for forward diffusion, we perturb the data with a linear combination of Gaussian noise and clean data as $\mathbf{x}_t = \sqrt{\bar{\alpha}_t}\mathbf{x} + \sqrt{1-\bar{\alpha}_t}\epsilon_t$ via a scheduler $\bar{\alpha}_t$ controlling the strength of added noise, here $t \in [0, T]$ and $\epsilon_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, note that when t reaches T, the perturbed data $\mathbf{x}_t$ become Gaussian noise. The training objective of the ϵ_θ is defined as the noise prediction MSE $\mathbb{E}_{t,\mathbf{x},\epsilon_t}[\|\epsilon_t - \epsilon_\theta(\mathbf{x}_t, t)\|_2^2]$. In sampling with diffusion models, Song et.al proposed DDIM (Song, Meng, and Ermon 2021) that generalized the DDPM sampling formulation as:

$$\mathbf{x}_{t-1} = \sqrt{\bar{\alpha}_{t-1}}\left(\frac{\mathbf{x}_t - \sqrt{1-\bar{\alpha}_t}\epsilon_\theta(\mathbf{x}_t, t)}{\sqrt{\bar{\alpha}_t}}\right) + \sqrt{1-\bar{\alpha}_{t-1}-\sigma_t^2}\epsilon_\theta(\mathbf{x}_t, t) + \sigma_t\epsilon_t.$$

The first term of the right-hand side of the equation represents direct clean sample estimation $\hat{\mathbf{x}}_0$ from noisy sample $\mathbf{x}_t$ which is derived from Tweedie's formula(Efron 2011). Therefore, the noise prediction can bridge with the score function via Tweedie's formula, where the denoising objective and score-matching objective are identical.

D Details of Our Proposed Algorithm

D.1 2-Wasserstein Distance Between Two Normal Distribution

Consider the normal distributions $\mathcal{N}_t := \mathcal{N}(\mu_t, \Sigma_t)$ and $\mathcal{N}_t^{\text{adv}} := \mathcal{N}(\mu_t^{\text{adv}}, \Sigma_t^{\text{adv}})$. Let $\Pi(\mathcal{N}_t, \mathcal{N}_t^{\text{adv}})$ denote a joint distribution over the product space $\mathbb{R}^n \times \mathbb{R}^n$. The 2-Wasserstein distance between $\mathcal{N}_t$ and $\mathcal{N}_t^{\text{adv}}$ is defined as:

$$\mathcal{W}_2^2(\mathcal{N}_t, \mathcal{N}_t^{\text{adv}}) = \min_{\pi \in \Pi(\mathcal{N}_t, \mathcal{N}_t^{\text{adv}})} \int \|\mathbf{f}_t - \mathbf{f}_t^{\text{adv}}\|_2^2 \mathrm{d}\pi(\mathbf{f}_t, \mathbf{f}_t^{\text{adv}}).$$

Using properties of the mean and covariance, we have the following identities:

$$\int \|\mu_t - \mu_t^{\text{adv}}\|_2^2 \mathrm{d}\pi(\mathbf{f}_t, \mathbf{f}_t^{\text{adv}}) = \|\mu_t - \mu_t^{\text{adv}}\|_2^2,$$

$$\int \|\mathbf{f}_t - \mu_t\|_2^2 \mathrm{d}\pi(\mathbf{f}_t, \mathbf{f}_t^{\text{adv}}) = \mathrm{trace}(\Sigma_t),$$

$$\int \|\mathbf{f}_t^{\text{adv}} - \mu_t^{\text{adv}}\|_2^2 \mathrm{d}\pi(\mathbf{f}_t, \mathbf{f}_t^{\text{adv}}) = \mathrm{trace}(\Sigma_t^{\text{adv}}),$$

$$\int (\mathbf{f}_t - \mu_t)^\top (\mathbf{f}_t^{\text{adv}} - \mu_t^{\text{adv}}) \mathrm{d}\pi(\mathbf{f}_t, \mathbf{f}_t^{\text{adv}}) = \mathrm{trace}\left(\mathbb{E}[(\mathbf{f}_t - \mu_t)(\mathbf{f}_t^{\text{adv}} - \mu_t^{\text{adv}})^\top\right).$$

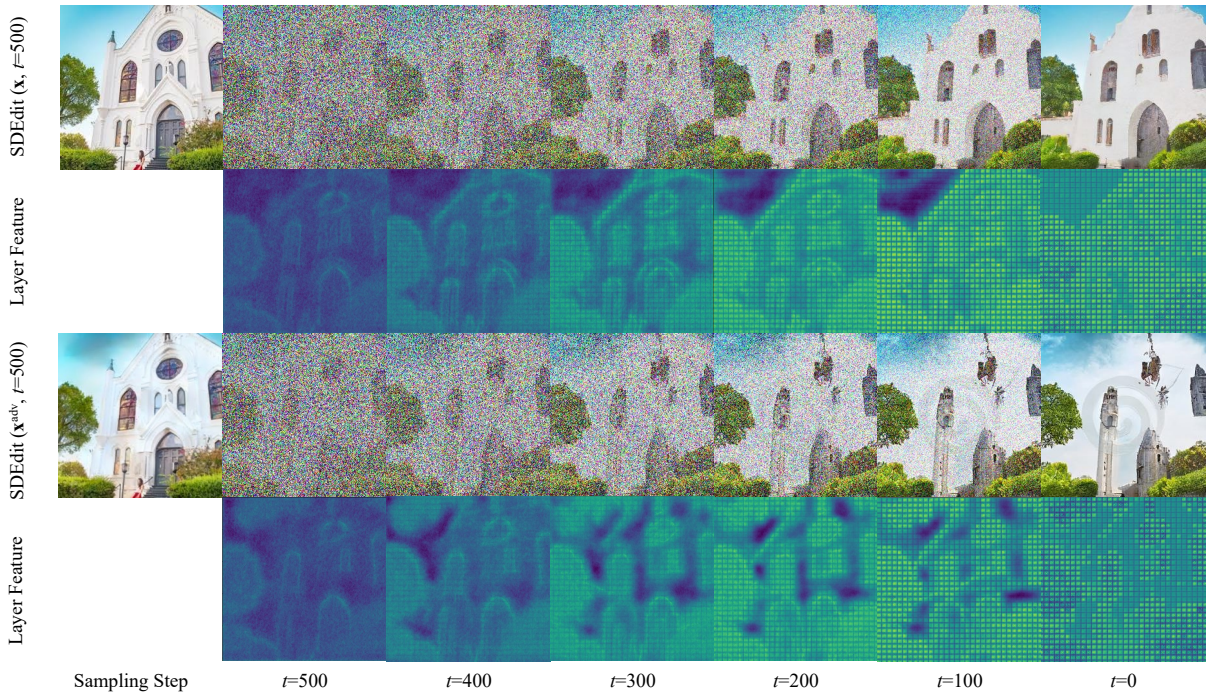

Figure 6: Qualitative example of corrupting feature representations in UNet: as the denoising process proceeds, the similarity of the feature map decreases, suggesting the representation is corrupted.

Thus, the 2-Wasserstein distance can be expressed as:

$$\mathcal{W}_2^2(\mathcal{N}_t, \mathcal{N}_t^{\text{adv}}) = \|\mu_t - \mu_t^{\text{adv}}\|_2^2 + \operatorname{trace}(\Sigma_t) + \operatorname{trace}(\Sigma_t^{\text{adv}}) - 2 \max_{J \succeq 0} \operatorname{trace}(C),$$

where J is the joint covariance matrix of $\mathcal{N}_t$ and $\mathcal{N}_t^{\text{adv}}$, defined as:

$$J = \begin{bmatrix} \Sigma_t & C \\ C^\top & \Sigma_t^{\text{adv}} \end{bmatrix},$$

and C is the covariance matrix between $\mathcal{N}_t$ and $\mathcal{N}_t^{\text{adv}}$:

$$C = \mathbb{E}\left[(\mathbf{f}_t - \mu_t)(\mathbf{f}_t^{\text{adv}} - \mu_t^{\text{adv}})^\top\right].$$

By the Schur complement, the problem can be formulated as a semi-definite programming (SDP) problem:

$$\begin{aligned} &\text{maximum} && \operatorname{trace}(C), \\ &\text{subject to} && \Sigma_t - C^\top (\Sigma_t^{\text{adv}})^{-1} C \succeq 0. \end{aligned}$$

The closed-form solution for C derived from the SDP is:

$$C = \Sigma_t^{\frac{1}{2}} (\Sigma_t^{\frac{1}{2}} \Sigma_t^{\text{adv}} \Sigma_t^{\frac{1}{2}})^{\frac{1}{2}} \Sigma_t^{-\frac{1}{2}}.$$

Finally, the closed-form solution for the 2-Wasserstein distance between the two normal distributions is given by:

$$\mathcal{W}_2^2(\mathcal{N}_t, \mathcal{N}_t^{\text{adv}}) = \|\mu_t - \mu_t^{\text{adv}}\|_2^2 + \operatorname{trace}(\Sigma_t) + \operatorname{trace}(\Sigma_t^{\text{adv}}) - 2(\Sigma_t^{\frac{1}{2}} \Sigma_t^{\text{adv}} \Sigma_t^{\frac{1}{2}})^{\frac{1}{2}}. \quad (6)$$

D.2 Alternating Optimization

Let $\mathbf{y} = \mathbf{x}^{\text{adv}}$, by Lagrange relaxation (Liu et al. 2023), the objective function can be expressed as:

$$F(\mathbf{x}, \mathbf{y}) = F_{\text{attack}}(\mathbf{x}, \mathbf{y}) + \lambda F_{\text{fidelity}}(\mathbf{x}, \mathbf{y}),$$

where $\lambda > 0$ is the Lagrange multiplier and F_{attack}, F_{fidelity} are defined as

$$\begin{aligned} F_{\text{attack}}(\mathbf{x}, \mathbf{y}) &= -\mathcal{L}_{\text{attack}}(\mathcal{F}(\mathbf{x}, t, \epsilon), \mathcal{F}(\mathbf{y}, t, \epsilon^{\text{adv}})), \\ F_{\text{fidelity}}(\mathbf{x}, \mathbf{y}) &= \max(\epsilon - \mathcal{L}_{\text{fidelity}}(\mathbf{x}, \mathbf{y}), \mathbf{0}). \end{aligned}$$

The optimization is carried out in an alternating manner as follows:

$$\mathbf{y}^{i+\frac{1}{2}} = \operatorname*{argmin}_{\mathbf{y}} \left(F_{\text{attack}}(\mathbf{x}, \mathbf{y}) + \lambda F_{\text{fidelity}}(\mathbf{x}, \mathbf{y}^i)\right), \quad (7)$$

$$\mathbf{y}^{i+1} = \operatorname*{argmin}_{\mathbf{y}} \left(F_{\text{attack}}(\mathbf{x}, \mathbf{y}^{i+\frac{1}{2}}) + \lambda F_{\text{fidelity}}(\mathbf{x}, \mathbf{y})\right). \quad (8)$$

To solve Equation 7, we employ the Fast Gradient Sign Method (FGSM) (Goodfellow, Shlens, and Szegedy 2015). The update is given by:

$$\mathbf{y}^{i+1/2} = \mathbf{y}^i - \gamma_{\text{attack}} \operatorname{sign}\left(\nabla_{\mathbf{y}^i} F_{\text{attack}}(\mathbf{x}, \mathbf{y}^i)\right).$$

For Equation 8, we utilize Gradient Descent, resulting in the following update:

$$\begin{aligned} \mathbf{y}^{i+1} &= \mathbf{y}^{i+\frac{1}{2}} - \tilde{\gamma}_{\text{fidelity}} \nabla_{\mathbf{y}^{i+\frac{1}{2}}} \lambda F_{\text{fidelity}}(\mathbf{x}, \mathbf{y}^{i+\frac{1}{2}}) \\ &= \mathbf{y}^{i+\frac{1}{2}} - \gamma_{\text{fidelity}} \nabla_{\mathbf{y}^{i+\frac{1}{2}}} F_{\text{fidelity}}(\mathbf{x}, \mathbf{y}^{i+\frac{1}{2}}). \end{aligned}$$

Figure 7: Qualitative example of different loss configurations. i. only semantic loss; ii. semantic loss and latent optimization; iii. semantic loss, $\mathcal{L}_{\text{fidelity}}$ and latent optimization.

Note that the gradient of F_{fidelity} can be derived as follows:

$$\nabla_{\mathbf{y}} F_{\text{fidelity}}(\mathbf{x}, \mathbf{y}) = \mathbb{I}_{\mathcal{C}'} \cdot \nabla_{\mathbf{x}_t^{\text{adv}}} \mathcal{L}_{\text{fidelity}}(\mathbf{x}, \mathbf{y}),$$

where $\mathbb{I}_{\mathcal{C}'}$ is indicator function with constraint $\mathcal{C} = \{\mathbf{y} \in \mathcal{M} \mid \mathcal{L}_{\text{fidelity}}(\mathbf{x}, \mathbf{y}) \leq \epsilon\}$.

Please note that after references, we also provide more results presented in Figures 7, 8, 9, and 5, please refer to subsequent pages.

D.3 AtkPDM Algorithm without Latent Optimization

Algorithm 2: AtkPDM

Input: Image to be protected $\mathbf{x}$, attack budget $\delta > 0$, and step size $\gamma_{\text{attack}}, \gamma_{\text{fidelity}} > 0$
Initialization: $\mathbf{x}^{\text{adv}} \leftarrow \mathbf{x}$, $L_{\text{attack}} \leftarrow \infty$
while L_{attack} not convergent **do**
  Sample timestep: $t \sim [0, T]$
  Sample noise: $\epsilon, \epsilon^{\text{adv}} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
  Compute original noisy sample: $\mathbf{x}_t \leftarrow \mathcal{F}(\mathbf{x}, t, \epsilon)$
  Compute adversarial noisy sample: $\mathbf{x}_t^{\text{adv}} \leftarrow \mathcal{F}(\mathbf{x}^{\text{adv}}, t, \epsilon^{\text{adv}})$
  Update $\mathbf{x}^{\text{adv}}$ by Gradient Descent: $\mathbf{x}^{\text{adv}} \leftarrow \mathbf{x}^{\text{adv}} - \gamma_{\text{attack}} \operatorname{sign}(\nabla_{\mathbf{x}^{\text{adv}}}(-\mathcal{L}_{\text{attack}}(\mathbf{x}_t^{\text{adv}}, \mathbf{x}_t)))$
  while $\mathcal{L}_{\text{fidelity}}(\mathbf{x}^{\text{adv}}, \mathbf{x}) > \delta$ **do**
    $\mathbf{x}^{\text{adv}} \leftarrow \mathbf{x}^{\text{adv}} - \gamma_{\text{fidelity}} \nabla_{\mathbf{x}^{\text{adv}}} \mathcal{L}_{\text{fidelity}}(\mathbf{x}^{\text{adv}}, \mathbf{x})$
  end while
end while
return $\mathbf{x}^{\text{adv}}$

E Limitations

While our method can deliver acceptable attacks on PDMs, its visual quality is still not directly comparable to the results achieved on LDMs, indicating room for further improvement. More generalized PDM attacks should be further explored.

F Societal Impacts

Our work will not raise potential concerns about diffusion model abuses. Our work is dedicated to addressing these issues by safeguarding images from being infringed.

Figure 8: Qualitative results of optimizing different fixed diffusion forward steps t^*.

Figure 9: Qualitative results compared to previous methods: our adversarial images can effectively corrupt the edited results without significant fidelity decrease. The same column shares the same random seed for fair comparison.