Towards Robust Defense against Customization via Protective Perturbation Resistant to Diffusion-based Purification

Wenkui Yang[1,2], Jie Cao[1], Junxian Duan[1], Ran He[1,2*]

[1]MAIS & NLPR, Institute of Automation, Chinese Academy of Sciences
[2]School of Artificial Intelligence, University of Chinese Academy of Sciences

yangwenkui.03@gmail.com, {jie.cao,junxian.duan}@cripac.ia.ac.cn, rhe@nlpr.ia.ac.cn

Abstract

Diffusion models like Stable Diffusion have become prominent in visual synthesis tasks due to their powerful customization capabilities, which also introduce significant security risks, including deepfakes and copyright infringement. In response, a class of methods known as protective perturbation emerged, which mitigates image misuse by injecting imperceptible adversarial noise. However, purification can remove protective perturbations, thereby exposing images again to the risk of malicious forgery.

In this work, we formalize the anti-purification task, highlighting challenges that hinder existing approaches, and propose a simple diagnostic protective perturbation named **AntiPure**. *AntiPure exposes vulnerabilities of purification within the "purification-customization" workflow, owing to two guidance mechanisms: 1) Patch-wise Frequency Guidance, which reduces the model's influence over high-frequency components in the purified image, and 2) Erroneous Timestep Guidance, which disrupts the model's denoising strategy across different timesteps. With additional guidance, AntiPure embeds imperceptible perturbations that persist under representative purification settings, achieving effective post-customization distortion. Experiments show that, as a stress test for purification, AntiPure achieves minimal perceptual discrepancy and maximal distortion, outperforming other protective perturbation methods within the purification-customization workflow.*

1. Introduction

Since the landmark developments of Denoising Diffusion Probabilistic Models (DDPMs) [15, 43] and Latent Diffusion Models (LDMs) [35], Diffusion Models (DMs) have virtually dominated every subtask of visual generation. This triumph can be attributed to the readily available open-source Stable Diffusion (SD) [48], which enables numer-
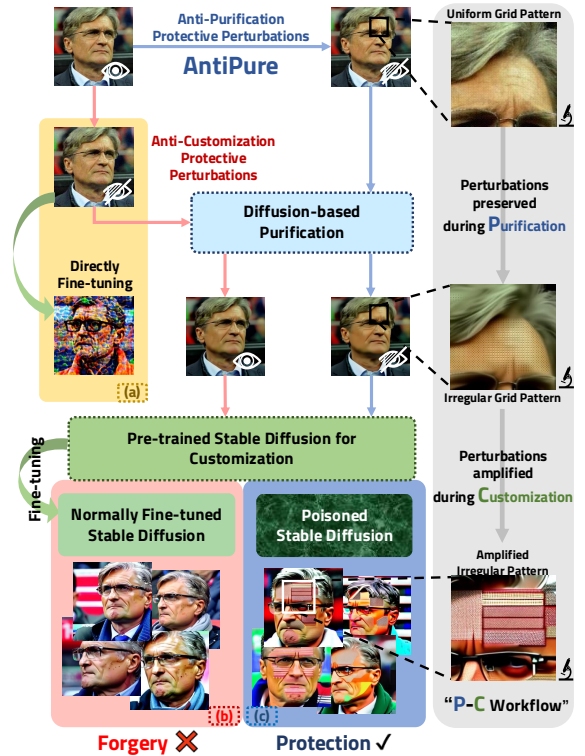

Figure 1. Overview diagram of "Purification-Customization" workflow. a). Protective perturbations use small noises to distort the outputs of fine-tuned diffusion models. b). However, existing methods can be removed by diffusion-based purification. c). We propose a simple diagnostic method called **AntiPure**, which achieves protective perturbations resistant to purification and makes customization outputs more distinguishable.

ous fine-tuning and editing techniques for user-friendly customization. However, these advances also pose risks, including the proliferation of deepfakes and infringements of portrait rights and intellectual property.

In response, recent studies [1, 24–26, 47, 51, 52, 57] adapt adversarial attacks [11, 22, 27] to DMs, creating "poisoned" samples that impede concept comprehension dur-

ing fine-tuning **Customization**. Specifically, these **Anti-Customization** approaches—collectively termed **Protective Perturbations**—employ white-box attacks to find the most adversarial perturbation noises within the l_∞-box, thereby distorting DMs' outputs.

Unfortunately, these adversarial protective perturbations can be nullified by diffusion-based **Purification** [31, 56, 61]. Such methods diffuse adversarial samples to a fixed timestep and then remove both adversarial and diffusion noises during denoising. As shown in Fig. 1, prior methods rarely consider how to preserve protective effects when purification precedes fine-tuning. This two-stage Purification-Customization (P-C) workflow therefore renders existing protective perturbations vulnerable and largely ineffective.

In this paper, we show the possibility of purification-resistant protective perturbations in representative P-C settings, despite the strong denoising capability of diffusion-based purification. First, we formalize how to achieve that within the P-C workflow. We observe that diffusion models—as a class of probabilistic models—produce outputs that can become highly unpredictable at the fine scale required by adversarial attacks, thereby diminishing the effectiveness of adaptive attacks. Based on this, rather than attempting to preserve the anti-customization perturbations during purification, we propose **anti-purification** perturbations that directly target the purification model to probe and expose its weaknesses. Through experiments, we analyze the differences between anti-customization and anti-purification, identifying three core characteristics of purification models that make anti-purification more challenging: 1) lack of vulnerable network components, 2) training-free frozen parameters, and 3) fixed high-timestep denoising.

Next, even under these stringent constraints, we propose a simple diagnostic method, **AntiPure**, which achieves an effective protective outcome by incorporating two additional types of guidance: **Patch-wise Frequency Guidance (PFG)** and **Erroneous Timestep Guidance (ETG)**. Since the priors on clean images embedded in frozen parameters prioritize low-frequency structures, the purification model lacks fine control over high-frequency details. Thus, PFG modulates the high-frequency components in each patch of the model's predicted "clean" image, leading to more localized perceptual discrepancy introduced by perturbation injection. Moreover, while overall structures are anchored by high-timestep denoising, ETG helps circumvent the timestep limitation. By minimizing the output distance across timesteps, ETG reduces the model's sensitivity to its timestep input and hinders its capacity to determine the appropriate actions at each step. Together, these mechanisms empower AntiPure to achieve both minimal perceptual discrepancy and maximal output distortion.

In brief, our contributions can be summarized as follows:
• We first formalize the requirements for effective protec-tive perturbations in the P-C workflow and propose anti-purification to target purification directly with a thorough analysis of the core challenges, overcoming the limitations of prior anti-customization perturbations.
• We propose a simple diagnostic method named AntiPure. AntiPure leverages Patch-wise Frequency Guidance and Erroneous Timestep Guidance to counter the aforementioned anti-purification challenges.
• Finally, we establish a benchmark to assess the effectiveness of prior anti-customization methods and our AntiPure within the P-C workflow. Experimental results show that AntiPure achieves both the lowest perceptual discrepancy and the highest output distortion as purification converges, outperforming existing methods.

2. Related Works

Customization with Stable Diffusion. With the rapid development of diffusion model [8, 14, 15, 43–46], a series of Text-to-Image (T2I) models [30, 33–35, 39] show exceptional potential for customized generation, where open-source Stable Diffusion (SD) [35] emerges as a community favorite. By various fine-tuning and conditional control techniques, pretrained models can further meet the needs for specific concepts and finer control. Textual Inversion [9] learns pseudo-words in the T2I models' embedding space. DreamBooth [38] incorporates class-specific prior preservation loss during full fine-tuning to mitigate forgetting. Following parameter-efficient fine-tuning (PEFT) [16], Custom Diffusion [21] modifies only the weights of cross-attention layers, while LoRA [17], adapted from large language models (LLMs), views concepts as the offset of parameters using rank decomposition matrices. For extra condition control, T2I-Adapter [29] and ControlNet [59] integrate additional guidance from other conditioning inputs.

Anti-Customization with Protective Perturbations. Advances in customization have raised significant concerns about deepfakes, privacy, and copyright. In this context, protective perturbation can serve as a potential solution to help prevent misuse and ensure authenticity. This can date back to the era of Generative Adversarial Networks (GANs) [10], when studies [18, 37, 54, 58] explored to distort the outputs of GAN-based image translation and editing [5, 12, 55, 62] by white-box attacks [11, 22, 23, 27].

For diffusion models, AdvDM [25] firstly employs Projected Gradient Descent (PGD) [27] to maximize the Latent Diffusion Model's training loss [55] using Monte Carlo estimation. This method is further extended in Mist [24] with the addition of textural loss. Glaze [41] safeguards artwork by attacking SD's encoder, while PhotoGuard [40] addresses unauthorized image inpainting by targeting both the encoder and UNet. Zhu et al. [63] utilize a GAN-based generator to create adversarial examples embedded with traceable watermarks. Anti-DreamBooth (Anti-DB)

[51] targets the fine-tuning process in DreamBooth by introducing a novel backpropagation surrogate to learn from both clean and partially adversarial examples. Building upon Anti-DB, SimAC [52] devises an adaptive greedy time interval selection. Zhao et al. [61] summarize the current challenges, providing a benchmark across fine-tuning methods and revealing the susceptibility of current techniques to purification. Concurrently, MetaCloak [26] meta-learns transformation-robust, transferable protections, while CAAT [57] perturbs cross-attention to obtain efficient, training-free perturbations. More recently, FastProtect [1] targets real-time deployment, and IDProtector [47] trains a one-pass encoder to defend against generation.

Adversarial Purification with Diffusion Models. Purification removes adversarial noises by regenerating or refining input images, with diffusion models gaining attention for their iterative denoising abilities [31, 53, 56]. DiffPure [31] employs an unconditional diffusion model to diffuse the adversarial sample over a selected timestep and then denoises it by solving the reverse-time SDE. DensePure [56] enhances the certified robustness of the pretrained classifier. However, these classifier-focused methods may not be sensitive to the high-resolution and perceptual consistency required for customization. GrIDPure [61] employs shorter timesteps diffusion with multiple iterations and overlapping grids to improve purification for customization.

3. Preliminaries

Here, we briefly introduce the necessary preliminaries. For more details, please refer to Appendix A.

Customization. The essence of customization is to fine-tune a model, pretrained on large-scale data, on a smaller, concept-specific set to capture that unseen concept, and Stable Diffusion (SD) [35] is a popular choice. Given input image x_0 and its text prompt y, the noise predictor UNet [36] ϵ_{θ_c} and the text encoder τ_{θ_c} with SD's parameters θ_c for customization are jointly optimized through:

$$\mathcal{L}_{ldm}(x_0; \theta_c) = \mathbb{E}_{\epsilon \sim \mathcal{N}(0,\mathbf{I}), t \sim \mathcal{U}(1,T)} \|\epsilon - \epsilon_{\theta_c}(z_t, t, \tau_{\theta_c}(y))\|_2^2, \tag{1}$$

where latent z_t is sampled through closed-form diffusion under reparameterization given VAE encoded $z_0 = \mathcal{E}(x_0)$.

Anti-customization utilizes adversarial attacks against generation, aiming to distort the concepts learned during fine-tuning by injecting protective perturbation δ^{adv}. For the optimal solution, this presents a saddle point problem:

$$\delta^{adv} = \arg\max_{\|\delta\|_\infty \leq \eta} \min_{\theta_c} \mathbb{E}_x \mathcal{L}_{ldm}(x_0 + \delta; \theta_c), \tag{2}$$

where $\delta^{adv} = x^{adv} - x_0$ is the adversarial perturbation restricted within the l_∞-ball of radius η. In practice, we often simplify Eq. (2) and maximize $\mathcal{L}_{ldm}$ to approximate the optimal δ^{adv} by white-box methods like I-FGSM [22, 23] or Projected Gradient Descent (PGD) [27].

Purification can remove adversarial noises while maintaining the global structures, thereby rendering protective perturbations ineffective. Pretrained unconditional diffusion models, such as DDPMs [15], can be inherently used for purification since the distributions of clean and adversarial samples converge over time during forward diffusion.

The pioneering work of diffusion-based purification, DiffPure [31], diffuses the input adversarial image at timestep t^p and denoises it back to a purified image. In simplified discrete DDPM form, this can be written as:

$$\mathrm{Pure}(x^{adv}) = \mathrm{Reverse}(\sqrt{\overline{\alpha}_{t^p}}(x^{adv}) + \sqrt{1 - \overline{\alpha}_{t^p}}\epsilon, t^p, 0; \theta_p), \tag{3}$$

where $\overline{\alpha}_{t^p} := \prod_{t=1}^{t^p}(1 - \beta_t)$ with β_t representing the diffusion variance. $\mathrm{Reserve}(\cdot)$ iteratively denoises the adversarial sample diffused at timestep t^p from higher timestep t^p to lower timestep 0 via learning-free sampling with frozen purification parameters θ_p. In the field of anti-customization, GrIDPure [61] adapts DiffPure to meet SD's requirements, transforming the longer denoising timestep into multiple shorter iterations, and achieving enhanced purified results.

4. Analysis

4.1. Anti-purification: Overall Formulation

For ideal perturbations resistant to purification, we first formalize our objective as follows:

$$\delta^{adv} = \arg\max_{\|\delta\|_\infty \leq \eta} \min_{\theta_c} \mathbb{E}_x \mathcal{L}_{ldm}(\mathrm{Pure}(x_0 + \delta); \theta_c), \tag{4}$$

where δ^{adv} is similarly optimized by maximizing $\mathcal{L}_{ldm}$, but with the purified input substituting the original one. However, direct backpropagation is computationally inefficient here, as purification generates extremely deep computation graphs over multiple iterations. Besides, unlike conventional adversarial attacks, anti-customization requires substantial massive memory overhead associated with SD, to which the size of classifiers is not comparable.

Alternatively, we decompose Eq. (4) into stages. Because the P-C workflow is serial, as long as one link in the procedure fails, effective fine-tuning cannot be realized. Interestingly, two opposing objectives can accomplish this:

$$\delta_{min}^{adv\prime} = \arg\min_{\|\delta\|_\infty \leq \eta} \|\mathrm{Pure}(x_0 + \delta) - (x_0 + \delta)\|_\infty, \text{ or} \tag{5}$$

$$\delta_{max}^{adv\prime} = \arg\max_{\|\delta\|_\infty \leq \eta} \|\mathrm{Pure}(x_0 + \delta) - (x_0 + \delta)\|_\infty. \tag{6}$$

By Eq. (5) combined with Eq. (2), we can approximate $\mathrm{Pure}(x) \approx x$, allowing Eq. (4) to degenerate into Eq. (2) even under purification. This follows the principle of adaptive attack and leads to a robust perturbation that can still target customization after purification. However, involving
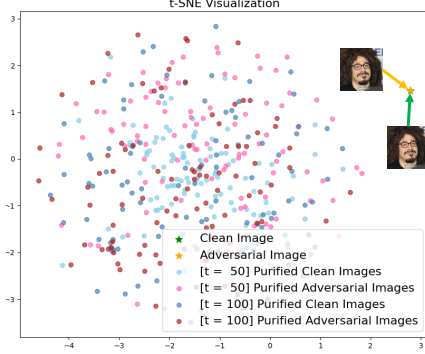
Figure 2. t-SNE [50] visualization (perplexity = 10) of 4×100 purified images obtained using DiffPure [31] with different timesteps for clean and adversarial images.

Figure 3. Attacks against DreamBooth [38] on UNet are much harder. Unlike vanilla pixel-space attacks ($a. \to d.$), latent-space attacks ($b. \to e.$) cannot target the vulnerable VAE encoder. Here, $d.$ (decoded from $e.$) is shown for visualization purposes only; in our experiments, we directly replace $b.$ with $e.$ during fine-tuning.

Figure 4. Mean Squared Errors of intermediate outputs between clean and adversarial samples across different UNet blocks at varying timesteps. See Appendix B.2 for details.

joint optimization of two objectives (Eq. (5) and Eq. (2)) makes the optimal solution hard to find as well. Additionally, probabilistic diffusion models may result in generated images that are almost completely uncontrollable at the fine scale required for adversarial noise. As shown in Fig. 2, the difference between the clean and adversarial images (which nearly overlap) is far smaller than the range of purified outputs, and the distributions of the purified clean and adversarial images converge as t^p increases. Therefore, we advocate for the alternative approach instead of treating purification as a special transformation and requiring the purified outputs to precisely reach the desired "adversarial region."

By Eq. (6), we resort to direct attacks against purification, i.e., **anti-purification**. We note that anti-purification is not intended as a wholesale replacement for anti-customization, but rather complements it as a preparatory step within diverse emerging workflows. In such scenarios, even if subsequent customization operates normally, the target concepts learned are still distorted from those of the original images during purification. This shifts the focus of our attack from the customization's LDMs to the DDPMs used in (DDPM-based) purification:

$$\mathcal{L}_{ddpm}(x_0; \theta_p) = \mathbb{E}_{\epsilon \sim \mathcal{N}(0,\mathbf{I}), t \sim \mathcal{U}(1,T)} \left\| \epsilon - \epsilon_{\theta_p}(x_t, t) \right\|_2^2. \tag{7}$$

Unfortunately, direct attacks via maximizing this DDPM training loss cannot achieve the same level of semantic structural distortion seen in anti-customization methods due to the inherent characteristics of purification itself. This makes anti-purification a more challenging task, with the difficulties encapsulated in the following three core reasons.

4.2. Anti-purification: Why harder?

Here, we report the conclusions of our small-scale experiments. For detailed settings, please refer to Appendix B.2.

4.2.1. Reason 1: Lack of Vulnerable Components

Attacks targeting LDMs/SD are easier due to their more vulnerable encoders. In contrast, the only component in
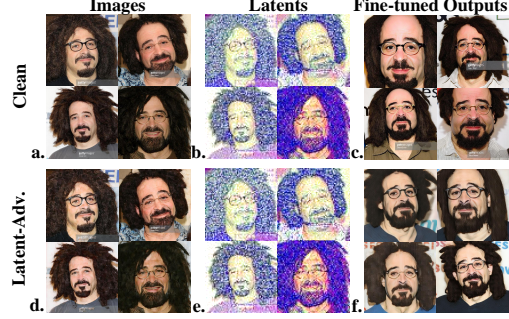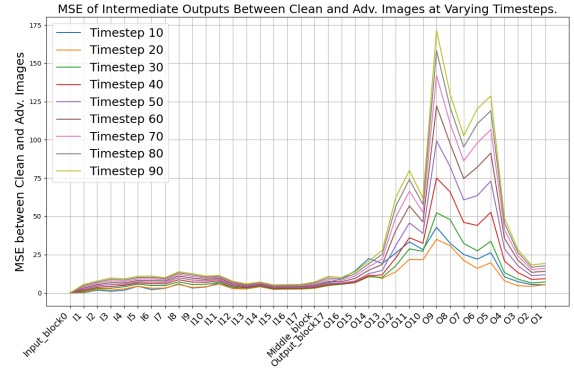
DDPMs, the UNet [36], is extremely robust.

Firstly, we modify Anti-DB's ASPL method [51] to conduct PGD attacks directly on Eq. (1) but in the latent space, thereby obtaining ($e.$) adversarial latents in Fig. 3 (rather than adversarial images for further encoding). As shown in Fig. 3, ($d.$) the decoded images from these latents exhibit unexpected stylistic transformations, and ($f.$) images generated by SD fine-tuned on these latents show minimal differences from ($d.$). This indicates that if PGD cannot leverage gradients from the vulnerable VAE encoder, the effectiveness of attacks is significantly reduced.

Also, we directly attack $\mathcal{L}_{ddpm}$. Fig. 4 illustrates the differences between clean and adversarial images in the outputs of various UNet blocks at different timesteps. Notice that there is minimal difference in the downsampling and intermediate blocks, while substantial deviations emerge in the middle part of the upsampling blocks. However, the outputs still converge at last, which may be attributed to the unique residual connections: the effect of adversarial noise naturally accumulates as the network's spatial structure iterates, while the shallower residual connections weaken this

Figure 5. Effectiveness of MasaCtrl [3] on adversarial images. The loss attack makes little difference except that the lower right image of $e.$ and $f.$ has slight noise in the background.

impact, forcing the perturbation to subside to lower levels, leading to a peak effect.

4.2.2. Reason 2: Training-free Frozen Parameters

It is both evident and important to note that, unlike anti-customization which targets fine-tuning by data poisoning, anti-purification targets a training-free editing task. In anti-purification, the pretrained purification model requires no additional adjustments, so adversarial samples cannot affect its benign priors embedded in frozen θ_p. Similar issues also arise in attacks on other training-free editing tasks. As shown in Fig. 5, the perturbations can reduce image quality in some cases for MasaCtrl [3], which also relies on vulnerable encoders. However, it does not achieve the same degradation level as perturbed fine-tuning.

4.2.3. Reason 3: Fixed High Timestep Denoising

During the denoising process, the low-frequency structural information of an image is largely determined at higher timesteps, while denoising at lower timesteps focuses on high-frequency, textural changes. Here, we notice that **the purification process can be viewed as a generation process where high-timestep denoising is fixed**. In cases where vulnerable components are absent and parameters are frozen, conducting a $\mathcal{L}_{ddpm}$-based attack for timesteps beyond t^p is not directly meaningful, and attempting to achieve semantic structural changes by adjusting the input at low timesteps is also unfeasible. Essentially, attacks on the purification itself are attacks restricted to high-frequency components at low timesteps.

5. Method: AntiPure

Building on the formulation in Sec. 4.1, we aim to generate the most adversarial input specifically targeting the purification model itself via Eq. (6). As highlighted in Sec. 4.2, achieving semantic structural distortion like anti-
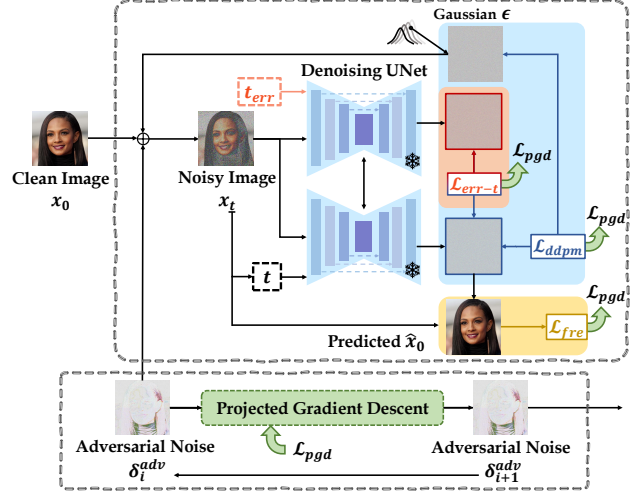

Figure 6. Overview diagram of AntiPure. AntiPure mitigates the negative impact of benign priors (Sec. 4.2.2) and limited timesteps (Sec. 4.2.3) on the attack by introducing Patch-wise Frequency Guidance $\mathcal{L}_{fre}$ and Erroneous Timestep Guidance $\mathcal{L}_{t-err}$.

customization is unfeasible for anti-purification. However, we can still raise the costs of customization and achieve protective perturbations that make the outputs from the P-C workflow more distinguishable. The overall attack process of our AntiPure is shown in Fig. 6.

5.1. Patch-wise Frequency Guidance

The priors on clean images embedded in the frozen network parameters allow the purification model to revert to a high-quality image distribution aligned with human intuition during the reverse denoising process. The overall outputs can often remain structurally high-quality even if a particular timestep is effectively attacked. However, unlike low-frequency semantic structures, consistency in high-frequency components is harder to guarantee, rendering them less controllable during purification. As shown in Fig. 7, although simply using $\mathcal{L}_{ddpm}$-attacks does not produce noticeable effects in the spatial domain, they introduce observable discrepancies in the frequency domain.

This phenomenon motivates shifting our focus to the frequency domain. A feasible solution is to modulate the high-frequency components where purification typically exerts weaker control, thereby causing the purified outputs to deviate from clean priors mainly in high-frequency bands. Also, due to the characteristics of human perception, modulating high-frequency components affects image semantic information less. For finer spatial modulation, we operate on smaller-resolution patches to localize perceptual discrepancy, forming our Patch-wise Frequency Guidance (PFG).

Specifically, given a clean image $x_0 \in \mathbb{R}^{C \times H \times W}$, a Gaussian noise ϵ, and an adversarial noise δ_i^{adv} at the i-th step of PGD, we diffuse the noisy adversarial sample x_t at
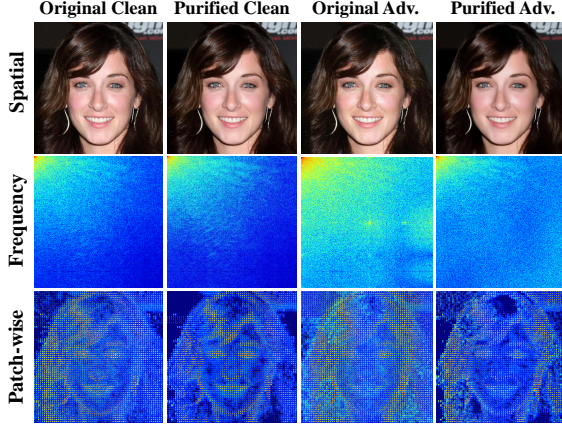
Figure 7. Differences in the spatial and frequency domains before and after DDPM-purification. Pseudocolor transformation is applied to the DCT spectrogram for better visualization.

timestep t as follows:

$$x_t = \sqrt{\overline{\alpha}_t}(x_0 + \delta_i^{adv}) + \sqrt{1 - \overline{\alpha}_t}\epsilon, \qquad (8)$$

where $\overline{\alpha}_t := \prod_{t'=1}^{t}(1 - \beta_{t'})$ with β_t representing the diffusion variance. Using the UNet's output $\epsilon_\theta(x_t, t)$, we approximate the predicted denoised image $\widehat{x}_0$ by:

$$\widehat{x}_0 = (x_t - \sqrt{1 - \overline{\alpha}_t}\epsilon_\theta(x_t, t))/\sqrt{\overline{\alpha}_t}. \qquad (9)$$

Next, Patch-wise Frequency Guidance operates on $\widehat{x}_0$ to track the UNet's gradients while emphasizing high-frequency components, which can be formalized as:

$$\mathcal{L}_{fre}(x_0; \delta^{adv}) = \sigma(\mathbb{E}_P \frac{4}{s^2} \sum_{m,n=\frac{s}{2}}^{s-1} \text{PatchDCT}(\widehat{x}_0, s)_{m,n}), \qquad (10)$$

where $\text{PatchDCT}(\cdot)$ (illustrated in 3rd row of Fig. 7) unfolds $\widehat{x}_0$ into P patches of size $s \times s$, and applies the Discrete Cosine Transform (DCT) to each patch. After DCT, the low-frequency component occupies the top-left corner of the spectrogram, while the high-frequency component lies in the bottom-right corner. Consequently, the bottom-right quarters of each patch are filtered, averaged, and then sigmoid-normalized by $\sigma(\cdot)$ to yield $\mathcal{L}_{fre}$.

In brief, $\mathcal{L}_{fre}$ aims to enhance the high-frequency components of the purification model's prediction after denoising across different timesteps, indirectly reinforcing the adversarial perturbation's high-frequency elements and creating a uniform grid pattern. Since the attack targets higher frequencies, localized structural information is minimally altered, ensuring greater perceptual consistency for humans.

5.2. Erroneous Timestep Guidance

As mentioned in Sec. 4.2.3, purification can essentially be viewed as a generative process where the high-timestep de-

noising has been fixed, meaning that the structure of images cannot be obviously altered. However, we can encourage the noise predictor's outputs at distinct timesteps to remain as close as possible. By injecting adversarial noise, we identify inputs for which the UNet struggles to select the appropriate actions across timesteps, which we realize via Erroneous Timestep Guidance (ETG).

Specifically, we select an erroneous timestep t_{err} as the input to the UNet to obtain a noise prediction of x_t at a higher timestep, and we minimize the discrepancy between the noise predicted at the erroneous time step $\epsilon_\theta(x_t, t_{err})$ and the correct prediction $\epsilon_\theta(x_t, t)$:

$$\mathcal{L}_{err-t}(x_0; \delta^{adv}) = - \|\epsilon_\theta(x_t, t_{err}) - \epsilon_\theta(x_t, t)\|_2^2. \quad (11)$$

5.3. Overall Attack

As shown in Fig. 6, our attack incorporates $\mathcal{L}_{fre}$ and $\mathcal{L}_{err-t}$ guidance with vanilla $\mathcal{L}_{ddpm}$. To balance the range of values across different training objectives, the total loss used for gradient ascent in PGD can be formalized as:

$$\mathcal{L}_{pgd}(x_0; \delta^{adv}) = \mathbb{E}_{\epsilon, t}\left(\mathcal{L}_{ddpm} + \lambda_1 e^{\overline{\alpha}_t - 1}\mathcal{L}_{fre} + \lambda_2 e^{\mathcal{L}_{err-t}}\right),$$
$$(12)$$

where the attack timestep $t \sim \mathcal{U}(1, t^p)$ is constrained within the purification step range t^p to avoid ineffective attacks. Both hyperparameters λ_1 and λ_2 are set to 0.5, with t_{err} fixed for convenience. The coefficient $e^{\overline{\alpha}_t - 1}$ scales $\mathcal{L}_{fre}$ to increase its impact as t decreases. Given the minimal MSE values, we apply an exponential function to $\mathcal{L}_{err-t}$ for slightly more aggressive optimization. Finally, PGD maximizes $\mathcal{L}_{pgd}$ to find the optimal δ^{adv}.

6. Experiments

6.1. Experimental Setup

Dataset. We utilize two facial datasets, CelebA-HQ [19] and VGGFace2 [4] for evaluation. Following Anti-DB [51], we select two subsets on these two datasets, each containing 50 IDs with 12 images at 512×512 resolution per ID, to ensure training for both Anti-DB and SimAC [52].

Baseline. We apply four advanced protective perturbation methods, including AdvDM [25], Mist [24], Anti-DB, and SimAC, where Anti-DB and SimAC require additional instances for perturbation generation. Thus, for each ID, we generate perturbation for 4 images and use another 4 images as instances for Anti-DB/SimAC.

Quantitative Metrics. For customization's output, we first employ **FID** [13], a general metric in generative tasks that quantifies the distance between the feature distributions of synthetic and real images using Inception v3 [49]. Additionally, we include metrics from Anti-DB [51]. Since our primary goal is to generate non-realistic faces, we employ

Dataset	Perturbation	FID↑	ISM↓ (FDFR)	BRISQUE↑
CelebA-HQ	AdvDM [25]	77.51	0.6561 (0.10)	31.33
	Mist [24]	70.23	0.6688 (0.07)	37.00
	Anti-DB [51]	78.84	0.6422 (0.10)	31.76
	SimAC [52]	67.37	0.6734 (0.09)	33.73
	AntiPure (Ours)	**81.15**	**0.6112** (0.10)	**43.60**
VGGFace2	AdvDM [25]	83.90	0.5923 (0.09)	37.42
	Mist [24]	78.34	0.5940 (0.07)	43.60
	Anti-DB [51]	90.29	0.5938 (0.06)	38.35
	SimAC [52]	75.21	0.6053 (0.09)	40.27
	AntiPure (Ours)	**90.77**	**0.5475** (0.05)	**46.01**

Table 1. Comparison of DreamBooth's [38] output quality for different perturbation methods following the P-C workflow.

Dataset	Workflow	FID↑	ISM↓ (FDFR)	BRISQUE↑
CelebA-HQ	AdvDM [25]	95.38	0.6302 (0.09)	38.20
	Mist [24]	85.09	0.6461 (0.07)	**40.91**
	Anti-DB [51]	104.18	0.6215(0.12)	38.18
	SimAC [52]	75.46	0.6487 (0.06)	38.77
	AntiPure (Ours)	**109.63**	**0.5839** (0.07)	40.01
VGGFace2	AdvDM [25]	105.43	0.5799 (0.07)	58.02
	Mist [24]	90.66	0.6046 (0.07)	62.22
	Anti-DB [51]	117.89	0.5723 (0.06)	58.56
	SimAC [52]	94.89	0.6018 (0.07)	59.99
	AntiPure (Ours)	**127.67**	**0.5428** (0.04)	**69.97**

Table 2. Comparison of LoRA's [17] output image quality for different perturbation methods following the P-C workflow.

Identity Score Matching (ISM) to evaluate the cosine similarity between the synthetic image's facial features and the real features of the corresponding identity using the popular ArcFace recognizer [6]. We also report Face Detection Failure Rate (FDFR) using the RetinaFace detector [7], measuring the ratio of undetectable faces excluded in ISM evaluation. This metric merely serves to provide ISM with a more comprehensive reference, as producing completely distorted faces after purification is, as previously discussed, unfeasible. Finally, we use an extra no-reference image quality assessment metric, **BRISQUE** [28], a classical and popular method for measuring generated image quality. For perturbation itself, a reconstruction metric, **LPIPS** [60] is provided to evaluate the difference in human perception. As we focus on disrupting the outputs of SD after purification, higher FID, lower ISM, and higher BRISQUE mean an increase in the adversarial effect. Besides, lower LPIPS means more imperceptible perturbation.

Configurations. For all perturbation methods, we use the same PGD training/purification/customization setup. In Sec. 6.2 and Sec. 6.3, we use GrIDPure [61] for purification, applying 2 rounds of 20 iterations with $t^p = 10$ to approximate convergence of the purification effect (validated in Sec. 6.4). In Sec. 6.5, we evaluate perceptual consistency with pretrained AlexNet[20]/VGG[42]. To verify the robustness of our method against common image processing techniques, we save the images with protective perturbations in their original input format, that is, the results on CelebA-HQ (.jpg) are all JPEG-compressed. We will elaborate on our experimental configurations in Appendix C.

Furthermore, for ablation studies on different perturbation configurations, black-box performance, and more visualization results, please refer to Appendix D.

6.2. Comparison with Baseline Methods

Quantitative Results. We first fine-tune SD on the purified images with different perturbations by DreamBooth, setting both the instance and inference prompt to "a photo of *sks* person." For fairness, the same set of 200 prior class images is shared between all perturbation methods. These quantitative results are shown in Tab. 1.

In Tab. 1, AntiPure achieves the best performance across all metrics on both datasets. After sufficient purification, the effectiveness of the previous methods is significantly reduced, but the effect of AntiPure can be better preserved. More importantly, while these methods depend on various customization configurations, AntiPure focuses solely on the relatively simpler purification process. On the other hand, due to the perturbed image degradation caused by JPEG compression, the results on CelebA-HQ are noticeably worse than those on VGGFace2, even when accounting for potential domain gaps.

Interestingly, SimAC shows an even greater drop in performance after purification. Compared to Anti-DB, the improved SimAC generates smoother edges around the perturbation patterns, rendering them more vulnerable. A similar distinction is also noted between AdvDM and Mist.

Qualitative Results. We show more results in Fig. 8. In our selected examples, the degree of output degradation intensifies from top to bottom. As previously discussed, after purification, perturbation methods cannot realistically achieve distortion at the level of overall semantic structure. AntiPure, however, does not aim to adjust the perturbations to disrupt the customization process to the maximum extent. Instead, it simply aims for the purification outputs to degrade, leveraging the characteristics of the fine-tuning process itself to amplify these artifacts. This approach may not achieve distortion of a person's ID information, but it is sufficient for humans to discern authenticity. For more visualization results, please refer to Appendix D.

6.3. Different Customization Method

To evaluate AntiPure's generalizability, we also apply another popular fine-tuning method, LoRA. Unlike

Figure 8. Visualization of DreamBooth's outputs after the P-C workflow.

Perturbation	Workflow	FID↑	ISM↓ (FDFR)	BRISQUE↑
None (Original)	C (Iter=0)	37.43	0.6935 (0.11)	15.86
Anti-DB [51]	P(Iter=10)-C	124.62	0.6020 (0.10)	32.74
	P(Iter=20)-C	84.83	0.6352 (0.09)	27.47
	P(Iter=30)-C	81.22	0.6473 (0.08)	29.33
	P(Iter=40)-C	77.30	0.6391 (0.09)	30.34
AntiPure (Ours)	P(Iter=10)-C	54.45	0.6362 (0.07)	40.27
	P(Iter=20)-C	59.97	0.6271 (0.08)	44.63
	P(Iter=30)-C	68.84	0.6075 (0.08)	47.68
	P(Iter=40)-C	78.21	0.5994 (0.09)	47.54

Table 3. Comparison of DreamBooth's [38] output image quality for different purification iterations following the P-C workflow on CelebA-HQ.

Perturbation	CelebA-HQ		VGGFace2	
	Alex-LPIPS↓	VGG-LPIPS↓	Alex-LPIPS↓	VGG-LPIPS↓
AdvDM [25]	0.2024	0.3061	0.2343	0.3920
Mist [24]	0.1470	**0.2759**	0.2208	0.5222
Anti-DB [51]	0.2019	0.3319	0.2726	0.4054
SimAC [52]	0.1754	0.3046	0.2146	0.4120
AntiPure (Ours)	**0.1392**	0.2843	**0.1758**	**0.3884**

Table 4. Comparison of Learned Perceptual Image Patch Similarity (LPIPS) [60] between adversarial images obtained by different perturbation methods and the original images.

DreamBooth, LoRA significantly reduces fine-tuning costs through low-rank decomposition. It can meanwhile incorporate the core class-specific prior preservation loss of DreamBooth. As shown in Tab. 2, our method still outperforms other perturbation methods, with a noticeable gap in the ISM metric compared to others, indicating that our method effectively reduces facial feature similarity.

6.4. Different Purification Configuration

Intuitively, as the purification iterations increase, protective perturbations are removed progressively, resulting in better fine-tuning output quality. To verify this, we select Anti-DB, the most robust method (apart from AntiPure) from previous experiments, and apply multiple rounds of purification with 10 iterations, followed by DreamBooth fine-tuning on CelebA-HQ. As shown in 3, the effectiveness of Anti-DB is indeed gradually diminished as the number of iterations increases, as expected, with ISM stabilizing around 20–40 iterations, indicating near convergence.

Unexpectedly, AntiPure, which focuses on countering the purification process, becomes increasingly robust with more iterations. Although its effectiveness is initially lower than that of Anti-DB with fewer iterations, AntiPure surpasses Anti-DB after approximately 30–40 iterations. This also demonstrates that the purification iteration settings in our previous experiments provide an accurate assessment of perturbation persistence at convergence.

6.5. Perceptual Consistency

In addition to preventing infringement and forgery, protective perturbations also need to avoid affecting human per-

ception, which is guaranteed by the hard constraint η. However, even within the same l_∞-ball, different perturbations can cause different perceptual shifts from clean images. To evaluate this, we select two commonly used backbones for LPIPS to assess the quality of different adversarial images under the same η constraint. As shown in Tab. 4, AntiPure achieves the smallest perceptual difference while providing the best attacks against the P-C workflow. This can be attributed to Patch-wise Frequency Guidance, which effectively avoids modifications to low-frequency information.

7. Conclusion

In this paper, we formalize anti-purification in the P–C workflow and present AntiPure, which incorporates two types of additional guidance. Across representative P-C settings, AntiPure remains effective despite the strong denoising capability of diffusion-based purification, advancing perturbation robustness for stronger prevention against deepfakes and infringement. However, as a simple diagnostic method, our current design assumes white-box access. Thus, we position AntiPure as a complement to anti-customization methods in diverse workflows, and leave improved black-box transfer as a direction for future work.

Acknowledgements

References

[1] Namhyuk Ahn, KiYoon Yoo, Wonhyuk Ahn, Daesik Kim, and Seung-Hun Nam. Nearly zero-cost protection against mimicry by personalized diffusion models. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pages 28801–28810, 2025. 1, 3

[2] Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *Int. Conf. Mach. Learn.*, pages 274–283. PMLR, 2018. 2

[3] Mingdeng Cao, Xintao Wang, Zhongang Qi, Ying Shan, Xiaohu Qie, and Yinqiang Zheng. Masactrl: Tuning-free mutual self-attention control for consistent image synthesis and editing. In *IEEE/CVF Int. Conf. Comput. Vis.*, pages 22560–22570, 2023. 5, 2

[4] Qiong Cao, Li Shen, Weidi Xie, Omkar M Parkhi, and Andrew Zisserman. Vggface2: A dataset for recognising faces across pose and age. In *IEEE Int. Conf. Autom. Face Gesture Recognit.*, pages 67–74, 2018. 6

[5] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pages 8789–8797, 2018. 2

[6] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pages 4690–4699, 2019. 7

[7] Jiankang Deng, Jia Guo, Evangelos Ververas, Irene Kotsia, and Stefanos Zafeiriou. Retinaface: Single-shot multi-level face localisation in the wild. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pages 5203–5212, 2020. 7

[8] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. In *Adv. Neural Inform. Process. Syst.*, pages 8780–8794, 2021. 2

[9] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion. In *Int. Conf. Learn. Represent.*, 2023. 2, 1

[10] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Adv. Neural Inform. Process. Syst.*, 2014. 2

[11] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *Int. Conf. Learn. Represent.*, 2015. 1, 2

[12] Zhenliang He, Wangmeng Zuo, Meina Kan, Shiguang Shan, and Xilin Chen. Attgan: Facial attribute editing by only changing what you want. *IEEE Trans. Image Process.*, 28 (11):5464–5478, 2019. 2

[13] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Adv. Neural Inform. Process. Syst.*, 2017. 6

[14] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *Adv. Neural Inform. Process. Syst. Worksh.*, 2021. 2

[15] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Adv. Neural Inform. Process. Syst.*, pages 6840–6851, 2020. 1, 2, 3

[16] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp. In *Int. Conf. Mach. Learn.*, pages 2790–2799. PMLR, 2019. 2

[17] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. In *Int. Conf. Learn. Represent.*, 2022. 2, 7, 1, 3, 4

[18] Hao Huang, Yongtao Wang, Zhaoyu Chen, Yuze Zhang, Yuheng Li, Zhi Tang, Wei Chu, Jingdong Chen, Weisi Lin, and Kai-Kuang Ma. Cmua-watermark: A cross-model universal adversarial watermark for combating deepfakes. In *AAAI Conf. Artif. Intell.*, pages 989–997, 2022. 2

[19] Tero Karras. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017. 6

[20] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Adv. Neural Inform. Process. Syst.*, pages 1097–1105, 2012. 7, 3

[21] Nupur Kumari, Bingliang Zhang, Richard Zhang, Eli Shechtman, and Jun-Yan Zhu. Multi-concept customization of text-to-image diffusion. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pages 1931–1941, 2023. 2

[22] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial machine learning at scale. In *Int. Conf. Learn. Represent.*, 2017. 1, 2, 3

[23] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial examples in the physical world. *arXiv preprint arXiv:1607.02533*, 2017. 2, 3, 1

[24] Chumeng Liang and Xiaoyu Wu. Mist: Towards improved adversarial examples for diffusion models. *arXiv preprint arXiv:2305.12683*, 2023. 1, 2, 6, 7, 8, 5

[25] Chumeng Liang, Xiaoyu Wu, Yang Hua, Jiaru Zhang, Yiming Xue, Tao Song, Zhengui Xue, Ruhui Ma, and Haibing Guan. Adversarial example does good: Preventing painting imitation from diffusion models via adversarial examples. In *Int. Conf. Mach. Learn.* PMLR, 2023. 2, 6, 7, 8, 5

[26] Yixin Liu, Chenrui Fan, Yutong Dai, Xun Chen, Pan Zhou, and Lichao Sun. Metacloak: Preventing unauthorized subject-driven text-to-image diffusion-based synthesis via meta-learning. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pages 24219–24228, 2024. 1, 3

[27] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *Int. Conf. Learn. Represent.*, 2018. 1, 2, 3

[28] Anish Mittal, Anush Krishna Moorthy, and Alan Conrad Bovik. No-reference image quality assessment in the spatial domain. *IEEE Trans. Image Process.*, 21(12):4695–4708, 2012. 7

[29] Chong Mou, Xintao Wang, Liangbin Xie, Yanze Wu, Jian Zhang, Zhongang Qi, and Ying Shan. T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models. In *AAAI Conf. Artif. Intell.*, pages 4296–4304, 2024. 2

[30] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. In *Adv. Neural Inform. Process. Syst.*, 2022. 2

[31] Weili Nie, Brandon Guo, Yujia Huang, Chaowei Xiao, Arash Vahdat, and Anima Anandkumar. Diffusion models for adversarial purification. In *Int. Conf. Mach. Learn.* PMLR, 2022. 2, 3, 4, 1

[32] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *Int. Conf. Mach. Learn.*, pages 8748–8763. PMLR, 2021. 2

[33] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *Int. Conf. Mach. Learn.*, pages 8821–8831. PMLR, 2021. 2

[34] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1 (2):3, 2022.

[35] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pages 10684–10695, 2022. 1, 2, 3

[36] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Int. Conf. Med. Image Comput. Comput. Assist. Interv.*, pages 234–241. Springer, 2015. 3, 4

[37] Nataniel Ruiz, Sarah Adel Bargal, and Stan Sclaroff. Disrupting deepfakes: Adversarial attacks against conditional image translation networks and facial manipulation systems. In *Eur. Conf. Comput. Vis. Worksh.*, pages 236–251. Springer, 2020. 2

[38] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pages 22500–22510, 2023. 2, 4, 7, 8, 1, 3, 5

[39] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. In *Adv. Neural Inform. Process. Syst.*, pages 36479–36494, 2022. 2

[40] Hadi Salman, Alaa Khaddaj, Guillaume Leclerc, Andrew Ilyas, and Aleksander Madry. Raising the cost of malicious ai-powered image editing. In *Int. Conf. Mach. Learn.* PMLR, 2023. 2, 4, 5

[41] Shawn Shan, Jenna Cryan, Emily Wenger, Haitao Zheng, Rana Hanocka, and Ben Y Zhao. Glaze: Protecting artists from style mimicry by {Text-to-Image} models. In *USENIX Security Symp.*, pages 2187–2204, 2023. 2

[42] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *Int. Conf. Learn. Represent.*, 2015. 7, 3

[43] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *Int. Conf. Mach. Learn.*, pages 2256–2265. PMLR, 2015. 1, 2

[44] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *Int. Conf. Learn. Represent.*, 2021. 2

[45] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Adv. Neural Inform. Process. Syst.*, 32, 2019.

[46] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *Int. Conf. Learn. Represent.*, 2021. 2

[47] Yiren Song, Pei Yang, Hai Ci, and Mike Zheng Shou. Idprotector: An adversarial noise encoder to protect against idpreserving image generation. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pages 3019–3028, 2025. 1, 3

[48] Stability AI. Stable diffusion, 2022. Available at: https://github.com/StabilityAI/stablediffusion. 1

[49] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pages 2818–2826, 2016. 6

[50] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *J. Mach. Learn. Res.*, 9(11), 2008. 4, 2

[51] Thanh Van Le, Hao Phung, Thuan Hoang Nguyen, Quan Dao, Ngoc N Tran, and Anh Tran. Anti-dreambooth: Protecting users from personalized text-to-image synthesis. In *IEEE/CVF Int. Conf. Comput. Vis.*, pages 2116–2127, 2023. 1, 3, 4, 6, 7, 8, 5

[52] Feifei Wang, Zhentao Tan, Tianyi Wei, Yue Wu, and Qidong Huang. Simac: A simple anti-customization method for protecting face privacy against text-to-image synthesis of diffusion models. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pages 12047–12056, 2024. 1, 3, 6, 7, 8, 5

[53] Jinyi Wang, Zhaoyang Lyu, Dahua Lin, Bo Dai, and Hongfei Fu. Guided diffusion model for adversarial purification. *arXiv preprint arXiv:2205.14969*, 2022. 3

[54] Lin Wang, Wonjune Cho, and Kuk-Jin Yoon. Deceiving image-to-image translation networks for autonomous driving with adversarial perturbations. *IEEE Robotics and Automation Letters*, 5(2):1421–1428, 2020. 2

[55] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pages 8798–8807, 2018. 2

[56] Chaowei Xiao, Zhongzhu Chen, Kun Jin, Jiongxiao Wang, Weili Nie, Mingyan Liu, Anima Anandkumar, Bo Li, and Dawn Song. Densepure: Understanding diffusion models for adversarial robustness. In *Int. Conf. Learn. Represent.*, 2023. 2, 3

[57] Jingyao Xu, Yuetong Lu, Yandong Li, Siyang Lu, Dongdong Wang, and Xiang Wei. Perturbing attention gives you more bang for the buck: Subtle imaging perturbations that efficiently fool customized diffusion models. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pages 24534–24543, 2024. 1, 3, 4, 5

[58] Chin-Yuan Yeh, Hsi-Wen Chen, Shang-Lun Tsai, and Sheng-De Wang. Disrupting image-translation-based deepfake algorithms with adversarial attacks. In *IEEE/CVF Winter Conf. Appl. Comput. Vis. Worksh.*, pages 53–62, 2020. 2

[59] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *IEEE/CVF Int. Conf. Comput. Vis.*, pages 3836–3847, 2023. 2

[60] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pages 586–595, 2018. 7, 8

[61] Zhengyue Zhao, Jinhao Duan, Kaidi Xu, Chenan Wang, Rui Zhang, Zidong Du, Qi Guo, and Xing Hu. Can protective perturbation safeguard personal data from being exploited by stable diffusion? In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pages 24398–24407, 2024. 2, 3, 7

[62] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *IEEE/CVF Int. Conf. Comput. Vis.*, pages 2223–2232, 2017. 2

[63] Peifei Zhu, Tsubasa Takahashi, and Hirokatsu Kataoka. Watermark-embedded adversarial examples for copyright protection against diffusion models. In *IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pages 24420–24430, 2024. 2

Towards Robust Defense against Customization via Protective Perturbation Resistant to Diffusion-based Purification

Supplementary Material

A. Details on Preliminaries

Here, we briefly review and supplement the preliminary knowledge from Sec. 3 to help our readers better understand the various tasks involved in this paper.

Fine-tuning Customization (Personalization). Fine-tuning methods aim to inject specific concepts into the pretrained SD for customization. Among them, DreamBooth (DB) [38] is widely studied for anti-customization. This approach not only minimizes $\mathcal{L}_{ldm}$ in few-shot scenarios but also incorporates a prior-preservation term to retain beneficial prior knowledge, thus mitigating forgetting. Its training objective can be formalized as:

$$
\mathcal{L}_{db}(x_0; \theta_c) = \mathcal{L}_{ldm}(x_0; \theta_c) \\
+ \lambda \underbrace{\mathbb{E}_{\epsilon', t'} \|\epsilon' - \epsilon_{\theta_c}(z_{t'}^{pr}, t', \tau_{\theta_c}(y^{pr}))\|_2^2}_{\text{Class-Specific Prior Preservation Loss}}, \quad (14)
$$

where the class prior image x^{pr} is generated by the pretrained model with class prompt y^{pr}, and $z_0^{pr} = \mathcal{E}(x^{pr})$ diffuses at timestep t' to form $z_{t'}^{pr}$. In $\mathcal{L}_{db}$, the loss term $\mathcal{L}_{ldm}$ employs instance prompts y of the form "a photo of [V][class noun]," where [V] acts as an identifier describing the target concept.

LoRA [17] is proposed to accelerate the optimization of large-scale pretrained models. It freezes the pretrained weights and injects trainable rank decomposition matrices into each layer, greatly reducing the number of trainable parameters for downstream tasks:

$$
W' = W_0 + A \cdot B, \quad (15)
$$

where W' and W_0 are fine-tuned and original weights, respectively, while $A \in \mathbb{R}^{m \times r}$ and $B \in \mathbb{R}^{r \times n}$ are low-rank matrices with rank $r \ll \min(m, n)$. LoRA can be used in conjunction with DB for efficient SD fine-tuning.

Adversarial Attack. In attacks against classifiers, whitebox methods like I-FGSM [22, 23] or PGD [27] are commonly used, which can be formalized as:

$$
x_{t+1}^{adv} = \Pi_{x_0, \eta} \left(x_t^{adv} + \alpha \cdot \text{sgn} \left(\nabla_{x_t^{adv}} \mathcal{L}(x_t^{adv}, y; \theta) \right) \right), \quad (16)
$$

where $\Pi_{x_0, \eta}(\cdot)$ restricts inputs within the l_∞-ball of radius η around x_0, $\text{sgn}(\cdot)$ represents the sign function, and α is the learning rate. $\mathcal{L}(x_t^{adv}, y; \theta)$ is the loss used by the classifier with parameters θ, where x_t^{adv} is the adversarial

sample at the t-th PGD step and y is the corresponding ground truth label. In brief, PGD iteratively finds the most adversarial noises for the model with parameters θ by maximizing the loss via gradient ascent.

Anti-customization. We explain the intuition behind the simplification that transforms our original maximizing objective from $\min_{\theta_c} \mathbb{E}_x \mathcal{L}_{ldm}(x_0 + \delta; \theta_c)$ to $\mathcal{L}_{ldm}(x_0 + \delta; \theta_c)$ as mentioned above. The key lies in the relationship between the model's training data x (of $\mathbb{E}_x$) and the adversarial data $x_0 + \delta^{adv}$. For optimal performance, the training set should encompass adequately trained adversarial samples. However, this creates a bootstrap paradox: fine-tuned θ_c is needed for optimal δ^{adv} while δ^{adv} is needed for optimal θ_c, which is why surrogate models fine-tuned on clean data are frequently employed for simplification.

In the context of fine-tuning methods like Textual Inversion [9], which make no change to the internal parameters of SD, such an issue exists no more in practice. For DB (full fine-tuning) and LoRA (PEFT), simply using a model fine-tuned on clean images as a surrogate leads to Fully-trained Surrogate Model Guidance (FSMG) formalized in Anti-DB [51]. A more promising alternative, also proposed by Anti-DB, is to iteratively introduce insufficient adversarial samples, generated at different PGD steps, into the surrogate model alongside clean images. This approach is referred to as Alternating Surrogate and Perturbation Learning (ASPL).

Purification. In its original paper, DiffPure [31] is introduced via Stochastic Differential Equation (SDE). Since we use the specialized DDPM-based purification model, and considering that SD is commonly implemented discretely, the introduction of diffusion-based purification in this paper is also written in the DDPM form. We present the more generalized SDE form from the original DiffPure here for both quick reference and rigor. For a Variance Preserving SDE (VP-SDE) where drift and diffusion coefficient are respectively defined as $f(x, t) := -\frac{\beta(t)}{2} x$ and $g(t) := \sqrt{\beta(t)}$, we first diffuse adversarial x^{adv} with a fixed timestep $t^p \in [0, 1]$ via:

$$
x(t^p) = \sqrt{\alpha(t^p)} x^{adv} + \sqrt{1 - \alpha(t^p)} \epsilon, \quad (17)
$$

where $\alpha(t) := e^{-\int_0^t \beta(s)ds}$, then we solve the reverse-time SDE to get the purified sample with an SDE solver sdeint:

$$
\text{Pure}(x^{adv}) = \text{sdeint}(x(t^p), f_{\text{rev}}, g_{\text{rev}}, \overline{w}, t^p, 0; \theta_p), \quad (18)
$$

where sdeint takes in six inputs: initial value $x(t^p)$, drift coefficient $f_{\text{rev}}(x, t) := -\frac{\beta(t)}{2}[x + 2s_{\theta_p}(x, t)]$, diffusion coefficient $g_{\text{rev}}(t) := \sqrt{\beta(t)}$, Wiener process $\overline{w}$, initial time t^p, and end time 0. In the discrete case, this whole purification process corresponds to the specialized DDPMs.

B. Details on Analysis

B.1. More Explanation on Overall Formulation

Due to the deepening of the computational graph during iterative purification denoising, full-gradient adaptive attacks lead to $\mathcal{O}(N)$ memory cost and may cause vanishing/exploding gradients. For a 2GB 256×256 unconditional DDPM purification model, fully tracking its training loss after only 5 consecutive denoising samplings requires up to 25GB memory overhead. For differentiability, DiffPure proposes the *adjoint method* to calculate full gradients of the reverse SDE with $\mathcal{O}(1)$ memory cost. However, this method of solving the augmented SDE does not reduce the time complexity. Backward Path Differentiable Approximation (BPDA) [2] is also a common approach, but the truly effective surrogate is hard to find.

Is it entirely infeasible to use full-gradient adaptive attacks? [61] mentions such a method for anti-customization, where DDIM [44] sampling strategy is utilized to ensure memory usage remains within an acceptable range. However, they report that this adaptive attack is not effective. To demonstrate the instability of purification diffusion models as probabilistic models, we set $\alpha = 0.005$, $\eta = \frac{16}{255}$, and perform a 100-step PGD attack on $\mathcal{L}_{ddpm}$. The resolution of the input images is 256×256. Subsequently, both the clean and the adversarial sample obtained from the attack are purified using DiffPure with $t^p = 50$ and $t^p = 100$, generating four sets of images, each containing 100 samples. The distributions of these sets are visualized using t-SNE [50] with perplexity set to 10, and the results are presented in Fig. 2. The convergence of the purified clean and adversarial samples motivates us to turn to the alternative by Eq. (6).

B.2. Experimental Details on the Reason Analysis

Reason 1: Lack of Vulnerable Components. Firstly, we modify Anti-DB's ASPL method to conduct PGD attacks directly in the latent space. We provide a more comprehensive experimental result here in Fig. 10, with the CLIP text encoder [32] taken into consideration. We set $\alpha = 0.005$, $\eta_z = \frac{16}{255}$, and perform a (20×5)-step PGD attack on $\mathcal{L}_{ldm}$. Fine-tuning steps per 5 PGD steps are set to 3. In the ASPL attack, we employ two configurations: one with a trainable text encoder (Latent-ASPL, trainable text encoder) and one with a frozen text encoder (Latent-ASPL, frozen text encoder), and the adversarial examples shown in Fig. 3 are obtained via the former. Actually, these two different configurations do not result in significant differences, whether

in the generated adversarial samples or in the outputs obtained after fine-tuning SD on the adversarial samples.

To avoid introducing additional noise during VAE decoding and to maintain consistency in the number of channels, we directly save the adversarial latents in ".pt" format and use them to replace the corresponding instance inputs in the DreamBooth training process.

During customization, we fine-tune SD v2.1 via Dream-Booth on these two kinds of adversarial samples. We also choose to either train or freeze the text encoder during fine-tuning. When jointly training the text encoder, we set the learning rate to $5e - 7$, and when freezing the text encoder, we set it to $5e - 6$ to ensure the capture of the target concept, with 500 steps of training for both fine-tuning configurations. The training instance prompt is "a photo of *sks* person", the class prompt is "a photo of person", and the inference prompt is "a photo of *sks* person". The training batch size is 2, with a prior loss weight of 1.0. The results are shown in Fig. 10.

Also, we directly attack $\mathcal{L}_{ddpm}$ using 200 randomly selected images in our datasets, resized to 256×256. We set $\alpha = 0.005$, $\eta = \frac{16}{255}$, and perform a 150-step PGD attack, with Monte-Carlo sampled timesteps limited in [1, 100]. Subsequently, the adversarial images and clean images are both fed into the UNet with time condition inputs ranging from 1 to 100. The purification model we use consists of 18 downsampling blocks, 1 middle block, and 18 upsampling blocks. We record MSE between the intermediate outputs of adversarial and clean images block by block under different time conditions. The average values across 200 images are computed, and the final results are presented in Fig. 4. It can be observed that, due to the increasing coefficient $\sqrt{1 - \overline{\alpha}_t}$, the differences between clean and adversarial images grow with longer timesteps.

Reason 2: Frozen Parameters with Benign Priors. The adversarial samples in Fig. 5 are generated using the $\mathcal{L}_{ldm}$-attack against SD v1.5. The configuration for generating protective perturbation is largely consistent with the setup used in the experiments above, conducted in the latent space. We set $\alpha = 0.005$, $\eta = \frac{16}{255}$, and perform a 100-step PGD attack. During editing, we employ MasaCtrl [3] combined with a pretrained T2I-Adapter [29], with the condition type set to "sketch." The significant attenuation of artifacts demonstrates that this direct attack on training objectives is not fully applicable to training-free tasks.

Reason 3: Fixed High Timestep Denoising. Perhaps the statement in Sec. 4.2.3, "the purification process can be viewed as a generation process where high-timestep denoising is fixed," is not sufficiently direct. To offer a more intuitive illustration of this process, we present a simple diagram. As depicted in Fig. 11, attacks on generation span the
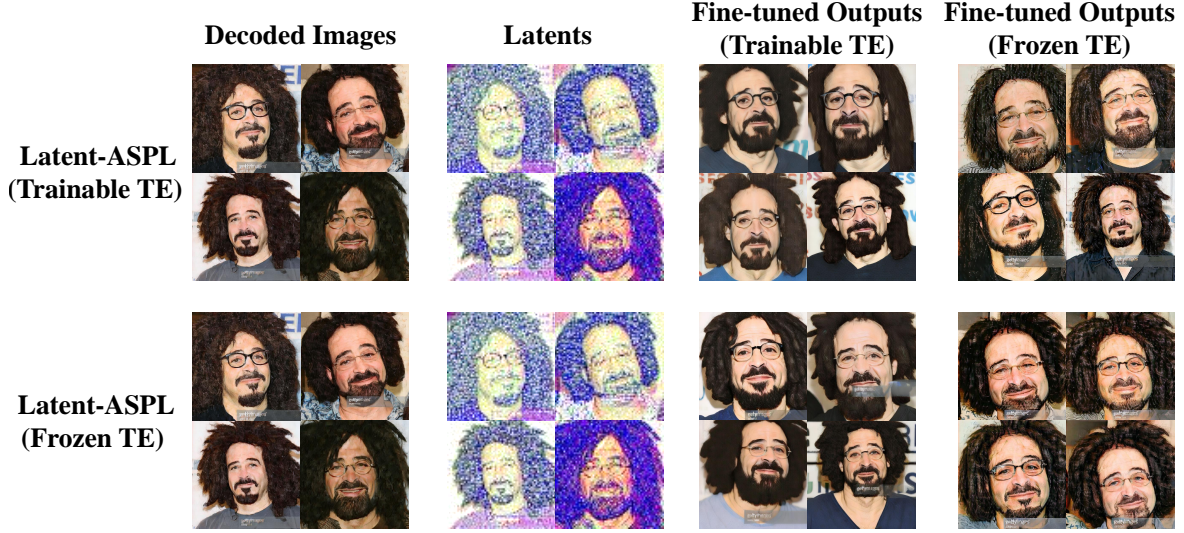
Figure 10. ASPL attacks [51] against SD in the latent space. In the ASPL attack, two configurations are used: trainable/frozen text encoder, corresponding to the two rows in the figure. Similarly, in the DreamBooth fine-tuning, the trainable/frozen text encoder configurations are also employed, corresponding to the last two columns.
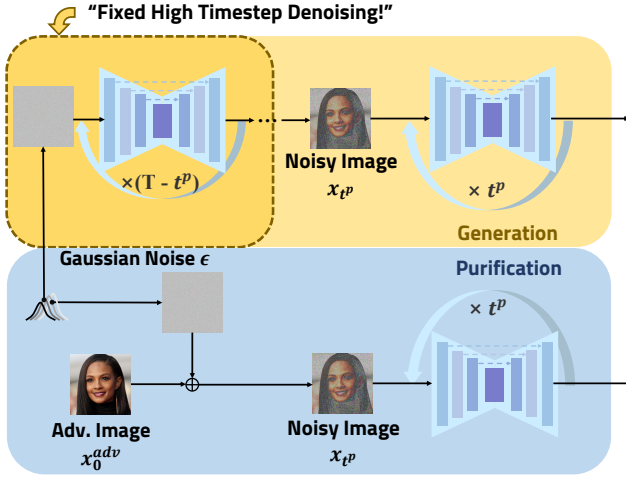

Figure 11. Why "the purification process can be viewed as a generation process where high-timestep denoising is fixed."

entire range from T (typically set to 1000) to 0. In contrast, attacks on purification are limited to a much smaller range, from t^p to 0, where the low-frequency structural information fixed during the "Fixed Higher Timestep Denoising" stage cannot be effectively altered.

C. Experimental Details

Here, we provide more configuration details used for the quantitative experimental evaluation. In Sec. 6.2 and Sec. 6.3, we use the same setup for all perturbation methods. In PGD attacks, we normalize images to $[-1, 1]$. Within this range, the noise budget η is set to $16/255$, the learning rate α is set to $5e^{-3}$, and the total PGD steps are set to 100 (20 ×

5 steps for Anti-DB and SimAC). In Sec. 6.5, we evaluate the perceptual consistency of different perturbation methods with pretrained AlexNet[20]/VGG[42].

The purification and fine-tuning settings are also kept consistent. In Sec. 6.2 and Sec. 6.3, we use GrIDPure [61] for purification, applying 2 rounds of 20 iterations with $t^p = 10$, $\gamma = 0.1$ to approximate convergence of the purification effect. In Sec. 6.4, we use a finer-grained purification configuration to explore when the purification effect approximately reaches convergence. Specifically, we apply 4 rounds of 20 iterations with $t^p = 10$, $\gamma = 0.1$, on the adversarial images obtained from Anti-DB and AntiPure. Totally, Tab. 3 uses 4 rounds × 10 iters, where *P(Iter=30)-C* equals 3×10, and so on. GrIDPure mitigates image degradation caused by purification via residual connections, allowing 4×10 to rely more on intermediate results while maintaining the same computational cost as 2×20. This leads to inconsistency between the results in Tab. 1 and the results of P(Iter=40)-C in Tab. 3. But overall, the computational overhead incurred by these two settings during the purification process is the same. Here, 20-iter is the default setting of GrIDPure.

For customization, in Sec. 6.2, we fine-tune the UNet and the text encoder jointly by DreamBooth [38] with batch size of 2 and learning rate of $5e^{-7}$ for 500 training steps. The training instance prompt is "a photo of *sks* person", the class prompt is "a photo of person", and the inference prompt is "a photo of *sks* person". We also set the prior loss weight to 1.0. In Sec. 6.3, we apply LoRA [17] with the same DreamBooth settings but set the learning rate to $5e^{-5}$. The rank is set to 4. For evaluation, 30 PNG images per ID are generated, which is also consistent with the configura-

Dataset	Objective	FID↑	ISM↓ (FDFR)	BRISQUE↑
CelebA-HQ	$\mathcal{L}_{ddpm}$	69.06	0.6293 (0.09)	42.45
	$\mathcal{L}_{ddpm}+\mathcal{L}_{fre}$	65.69	0.6253 (0.08)	42.84
	$\mathcal{L}_{ddpm}+\mathcal{L}_{err-t}$	74.42	0.6489 (0.10)	37.01
	AntiPure (Ours)	**81.15**	**0.6112** (0.10)	**43.60**
VGGFace2	$\mathcal{L}_{ddpm}$	76.32	0.5958 (0.07)	39.42
	$\mathcal{L}_{ddpm}+\mathcal{L}_{fre}$	74.90	0.5644 (0.07)	45.57
	$\mathcal{L}_{ddpm}+\mathcal{L}_{err-t}$	76.75	0.5901 (0.06)	40.75
	AntiPure (Ours)	**90.77**	**0.5475** (0.05)	**46.01**

Table 6. Ablation Study on DreamBooth's [38] output quality for different AntiPure guidance following the Purification-Customization (P-C) workflow.

Dataset	Objective	FID↑	ISM↓ (FDFR)	BRISQUE↑
CelebA-HQ	$\mathcal{L}_{ddpm}$	93.79	0.6176 (0.05)	42.19
	$\mathcal{L}_{ddpm}+\mathcal{L}_{fre}$	81.32	0.5848 (0.05)	42.24
	$\mathcal{L}_{ddpm}+\mathcal{L}_{err-t}$	92.63	0.6177 (0.09)	**43.22**
	AntiPure (Ours)	**109.63**	**0.5839** (0.07)	40.01
VGGFace2	$\mathcal{L}_{ddpm}$	93.10	0.5859 (0.08)	61.79
	$\mathcal{L}_{ddpm}+\mathcal{L}_{fre}$	110.87	0.5556 (0.06)	66.01
	$\mathcal{L}_{ddpm}+\mathcal{L}_{err-t}$	102.24	0.5717 (0.06)	61.10
	AntiPure (Ours)	**127.67**	**0.5428** (0.04)	**69.97**

Table 7. Ablation Study on LoRA's [17] output quality for different AntiPure guidance following the Purification-Customization (P-C) workflow.

Dataset	Transformation	FID↑	ISM↓ (FDFR)	BRISQUE↑
VGGFace2	Crop-Scale	152.47	0.4805 (0.34)	53.60
	Rotation	92.00	0.5550 (0.05)	45.14
	None (Ours)	**90.77**	**0.5475** (0.05)	**46.01**

Table 8. Comparison of DreamBooth's [38] output quality on VGGFace2 for different transformations on AntiPure's outputs.

tions used in Anti-DB and SimAC.

D. More Experimental Results

D.1. Ablation Study

Our proposed AntiPure incorporates two kinds of additional guidance to address the inherent challenges of the anti-purification task: 1) Patch-wise Frequency Guidance and 2) Erroneous Timestep Guidance. In the ablation study, we gradually remove this guidance to validate the effectiveness of our method.

We use the same attack/purification/customization experimental configurations in Sec. 6.2 and Sec. 6.3 to perform the corresponding DreamBooth and LoRA fine-tuning on CelebA-HQ and VGGFace2, but with different attack targets. Specifically, our attack targets include: 1) $\mathcal{L}_{ddpm}$, 2) $\mathcal{L}_{ddpm}+\mathcal{L}_{fre}$, 3) $\mathcal{L}_{ddpm}+\mathcal{L}_{err-t}$, and we compare these results with the full AntiPure, i.e., 4) $\mathcal{L}_{ddpm}+\mathcal{L}_{fre}+\mathcal{L}_{err-t}$. The DreamBooth fine-tuning results are shown in Tab. 6, and the LoRA results are shown in Tab. 7.

It is evident that AntiPure, which combines both types of guidance, achieves the best overall performance across various metrics, datasets, and fine-tuning methods, resulting in the most significant output distortion. This represents a clear improvement over the original $\mathcal{L}_{ddpm}$-based attack. Additionally, it can be observed that among the single guidance methods, $\mathcal{L}_{fre}$ is more effective than $\mathcal{L}_{err-t}$. In fact, using $\mathcal{L}_{err-t}$ for extra guidance alone shows limited impact. However, it helps confuse the model across different time steps, thereby disrupting the frequency characteristics of the predicted noise, providing a better foundation for $\mathcal{L}_{fre}$ guidance. This is particularly evident in FID, where AntiPure sees an obvious improvement when both types of guidance are combined.

In other words, the combination of these two guidance types is not merely an additive process but achieves a synergistic "$1+1>2$" effect. Actually, the timestep inputs of the diffusion model's UNet can affect the frequency repre-

sentation of the predicted noise, allowing $\mathcal{L}_{err-t}$ to be interpreted on the frequency domain like $\mathcal{L}_{fre}$. With both involved, the high-frequency components intensified by $\mathcal{L}_{fre}$ are primarily induced by erroneous high timesteps rather than real ones. Thus, the introduction of $\mathcal{L}_{err-t}$ can indirectly enhance $\mathcal{L}_{fre}$ itself, and vice versa.

D.2. Transformation Robustness

As suggested by the reviewer, we apply *Crop-Scale* (Center-Crop $\times 3/4$ side length) and *Rotation* (randomly $[0°, 15°]$) to anti-purification samples created by AntiPure, ensuring that the same transformations are applied to the original ones for fair evaluation. As shown in Tab. 8, AntiPure demonstrates robustness to *rotation*, while *crop-scale* amplifies the artifacts, leading to significantly improved performance.

D.3. More Baselines

As suggested by the reviewer, we include PhotoGuard [40] and CAAT [57] for additional comparison. We adopt the *img2img* attack pipeline for PhotoGuard, as it resembles purification more than the *inpainting* pipeline. However, as shown in Tab. 9, PhotoGuard's perturbations tend to be easily purified due to their blurred boundaries. In contrast, CAAT's perturbation closely resembles that of Anti-DB, leading to comparable robust performance.

Dataset	Perturbation	FID↑	ISM↓ (FDFR)	BRISQUE↑
	PhotoGuard [40]	72.25	0.6061 (0.07)	43.07
VGGFace2	CAAT [57]	89.07	0.5854 (0.07)	38.21
	AntiPure (Ours)	**90.77**	**0.5475** (0.05)	**46.01**

Table 9. Comparison with additional baselines on VGGFace2.

Dataset	λ_1	λ_2	t_{err}	FID↑	ISM↓ (FDFR)	BRISQUE↑
	0.25	0.75	999	96.33	0.5431 (0.06)	41.50
VGGFace2	0.50	0.50	700	90.50	0.5490 (0.04)	48.34
	0.75	0.25	999	87.81	0.5586 (0.05)	43.02
	0.50	0.50	999	**90.77**	**0.5475** (0.05)	**46.01**

Table 10. DreamBooth's [38] output quality on VGGFace2 for different hyperparameter configurations.

Fine-tuning	Perturbation	FID↑	ISM↓ (FDFR)	BRISQUE↑
	AdvDM [25]	82.10	0.5798 (0.06)	26.99
	Mist [24]	77.33	0.5797 (0.04)	32.58
DreamBooth	Anti-DB [51]	83.95	0.5686 (0.06)	27.68
	SimAC [52]	76.73	0.5762 (0.05)	26.44
	AntiPure (Ours)	**89.33**	**0.5165** (0.03)	**62.88**
	AdvDM [25]	106.48	0.5697 (0.05)	44.96
	Mist [24]	91.33	0.5731 (0.06)	55.27
LoRA	Anti-DB [51]	**115.34**	0.5591 (0.05)	46.11
	SimAC [52]	92.57	0.5622 (0.05)	45.41
	AntiPure (Ours)	112.90	**0.5101** (0.05)	**74.82**

Table 11. Comparison of DreamBooth/LoRA's [38] Stable Diffusion v1.5 output quality on VGGFace2 for different perturbation methods following the P-C workflow.

D.4. Hyperparameter Sensitivity

Originally, the selection of λ_1 and λ_2 was based on balancing the magnitude of loss components, while t_{err} was chosen to be as large as possible to maximize its effect. Here, as the reviewer suggested, we conduct a simple grid search over these three hyperparameters. As shown in Tab. 10, different metrics exhibit varying degrees of sensitivity to each parameter. Notably, the impact of t_{err} is relatively smaller compared to those of λs, while the ISM—the primary metric for identity preservation—remains largely stable across all settings. This suggests that AntiPure exhibits a certain degree of robustness with respect to its hyperparameter configurations.

D.5. Black-Box Performance

All previous experiments are conducted on SD v2.1, as recommended by Anti-DB and SimAC. However, AdvDM and Mist only support SD v1.x. We note that after sufficient purification, the effects of these perturbation methods almost completely disappear, making the distinction between SD versions insignificant.

To evaluate the performance of perturbation methods under a black-box scenario with mismatched models, and to ensure an absolutely fair SD version for all methods, we fine-tune SD v1.5 on the purified adversarial images from VGGFace2. The results are shown in Tab. 11. The similar performance observed preliminarily supports our hypothesis that "SD versions have negligible influence." Also, AntiPure still demonstrates the best overall performance.

D.6. More visualization

We provide more visualization results in Figs. 12 to 15 for qualitative evaluation. Please refer to the captions of each figure for detailed explanations. **We strongly recommend zooming in** on the following visualizations to better iden-

tify these artifacts.

It can be observed that the effects of other protective perturbation methods almost entirely vanish after sufficient purification. However, AntiPure ensures the presence of detectable artifacts, which are concentrated in the human facial regions (excluding the eyes). At lower levels of semantic distortion, these artifacts appear as unnatural high-frequency speckled regions, while more prominent artifacts manifest as patches of abnormal textures.

Furthermore, the effects of different perturbation methods on human visual perception (*iter = 0*, i.e., no purification) in Figs. 14 and 15 are also consistent with the LPIPS comparison in Tab. 4. Even under the same noise budget, Anti-DB and CAAT perturbations are more noticeable, often exhibiting blocky color artifacts. AntiPure, however, relies on frequency-domain modulation and generates samples visually closer to the original image.

Figure 12. Comparison of DreamBooth's outputs on CelebA-HQ for different perturbation methods following the Purification-Customization (P-C) workflow.

Figure 13. Comparison of DreamBooth's outputs on VGGFace2 for different perturbation methods following the Purification-Customization (P-C) workflow.

Figure 14. Comparison of GrIDPure's outputs at different iterations on VGGFace2 for different perturbation methods. Here *Iter=0* means no purification is adopted after adversarial samples are generated.

Figure 15. Comparison of GrIDPure's outputs at different iterations on VGGFace2 for different perturbation methods. Here *Iter=0* means no purification is adopted after adversarial samples are generated.