# SimAC: A Simple Anti-Customization Method for Protecting Face Privacy against Text-to-Image Synthesis of Diffusion Models

Feifei Wang[1,2,*] Zhentao Tan[2,1], Tianyi Wei[1], Yue Wu[2], Qidong Huang[1,†]
[1]University of Science and Technology of China [2]Alibaba Cloud
{wangfeifei@, tzt@, bestwty@, hqd0037@}mail.ustc.edu.cn matthew.wy@alibaba-inc.com

## Abstract

*Despite the success of diffusion-based customization methods on visual content creation, increasing concerns have been raised about such techniques from both privacy and political perspectives. To tackle this issue, several anti-customization methods have been proposed in very recent months, predominantly grounded in adversarial attacks. Unfortunately, most of these methods adopt straightforward designs, such as end-to-end optimization with a focus on adversarially maximizing the original training loss, thereby neglecting nuanced internal properties intrinsic to the diffusion model, and even leading to ineffective optimization in some diffusion time steps. In this paper, we strive to bridge this gap by undertaking a comprehensive exploration of these inherent properties, to boost the performance of current anti-customization approaches. Two aspects of properties are investigated: 1) We examine the relationship between time step selection and the model's perception in the frequency domain of images and find that lower time steps can give much more contributions to adversarial noises. This inspires us to propose an adaptive greedy search for optimal time steps that seamlessly integrates with existing anti-customization methods. 2) We scrutinize the roles of features at different layers during denoising and devise a sophisticated feature-based optimization framework for anti-customization. Experiments on facial benchmarks demonstrate that our approach significantly increases identity disruption, thereby protecting user privacy and copyright. Our code is available at: https://github.com/somuchtome/SimAC.*

## 1. Introduction

Latent Diffusion model (LDM) [12, 31, 32] has been recently proven as a strong paradigm for photorealistic visual content generation. The emergence of open-source Stable Diffusion encourages users to explore creative possibilities with LDMs. Users only need to provide several images representing the same subject along with a rare identifier to customize their diffusion models [7, 13, 22, 28]. After fine-tuning the model or optimizing the text embedding of rare identifiers, it can flexibly generate high-quality images containing the specified object.

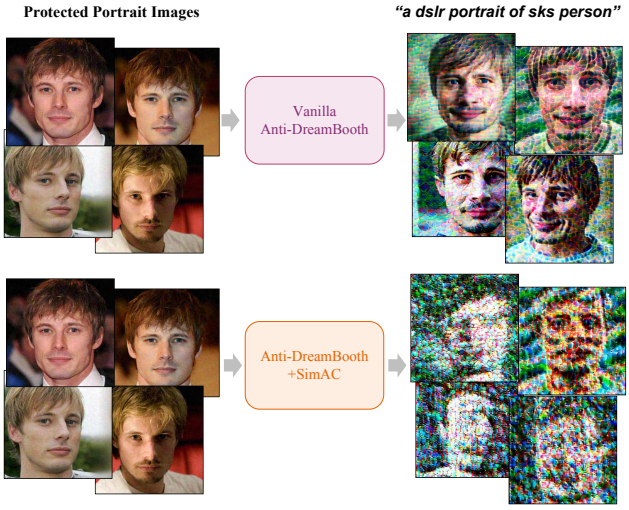
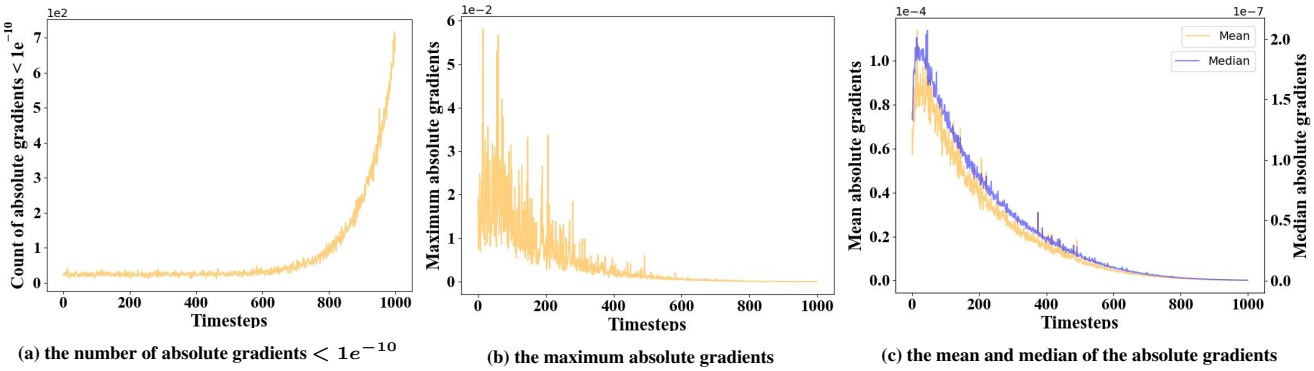



Figure 1. Comparison between Anti-DreamBooth before and after adding SimAC. Our method further boosts its ability to de-identity.

The convenience of customizing large-scale text-to-image models also allows the malicious users to generate forgery images that violate the truth. Some of them may use such technique to steal others' painting styles and generate new painting content without permission. While some of them may collect one's portrait photos that are published on the social platform, and generate fake images of this person through customization. In other words, the lawbreakers can easily produce fake news that contains celebrity photos [2], greatly misleading the public. The infringement poses threats to user privacy and intellectual property. Hence, it is essential to devise effective countermeasures to safeguard users against such malicious usage.

*Work done during an internship in Alibaba Cloud.
†Corresponding author.




(a) the number of absolute gradients $< 1e^{-10}$ (b) the maximum absolute gradients (c) the mean and median of the absolute gradients

Figure 2. Distribution of Anti-DreamBooth attack gradients on different diffusion timesteps, where (a) counts the number of absolute gradients below the threshold $1e^{-10}$ at each timestep, (b) presents the maximum absolute gradients at different timesteps, and (c) demonstrates how the mean and median of the absolute gradients change over timesteps. Apparently, the absolute gradients shows great discrepancy on the varying timesteps and nearly zero values appear at large timesteps, which leads to ineffective noise optimization.

Current anti-customization methods are generally based on adversarial attack [9, 33]. AdvDM [23] is the pioneering work that uses adversarial noise to protect user images from being customized by diffusion models. It ingeniously combines the diffusion model with adversarial samples, firstly achieving user privacy protection. Anti-DreamBooth [36] further enhances the protective effect by employing alternate training. Both of them randomly select the timesteps from (0, *max denoising steps*) regarding the noise added to LDM latent, and directly employ the maximization of the LDM training loss as the optimization objective.

However, as showcased in Figure 2, we notice that the gradient of the perturbed images is quite small and even **zero** when the randomly sampled timesteps are relatively large, *i.e.*, the noisy latent is closer to Gaussian noise. This implies that within the limited steps of an adversarial attack, a large portion of steps are ineffective since the perturbed image cannot optimize the objective with such zero gradients, leading to a decrease in both protection effectiveness and efficiency. Hence, the images protected by these current methods, when customized, still allow the model to capture many details from user-input images and leak the user's privacy, as shown in Figure 1.

The fundamental reason is that these methods fail to combine the adversarial attacks with properties inherent in diffusion models. We aim to conduct an in-depth analysis of why the gradients of perturbed images have such discrepancy at different time steps and how diffusion models perceive input images at different intermediate layers. Then, based on the in-depth analysis, we propose improvements to existing customization methods from both the temporal and feature dimensions.

To observe the relationship between timesteps and the gradient of the perturbed images, we first reconstruct images predicted from noisy latents at different time steps and compare them with the input images. Considering regular adversarial noises mainly affect the high frequency of images, our comparison is conducted in the frequency domain, aiming at investigating whether the model's perception lies on lower-level or higher-level. We observe that, the model focuses on the higher frequency components of images when selecting smaller time steps, and *vice versa*. Therefore, introducing adversarial noises at larger timesteps is ineffective, since the subtle changes perturbed on images can hardly affect the low-frequency of the generated images. To improve the effectiveness, we propose an adaptive timestep selection method to find optimal time intervals, where we iteratively update the range of selection in a greedy way. To explore the effect of different layers' features in the U-Net decoder during denoising, we employ PCA (Principal Component Analysis) to visualize them and show the discrepancy. We clearly find that within the decoder, the feature extraction gradually shifts from low-frequency to high-frequency as the layer goes deeper. It indicates the higher layers concentrates more on the texture of images, while the lower layers focus on the structure. Consequently, we select the features representing high-frequency information during optimization, since regular adversarial noises concentrate on the high frequency of images. And we construct the feature interference loss and integrate it with the diffusion denoising loss as the objective function, to improve the ability of identity interference.

Our main contributions are as follows:

- We reveal the inadequate optimization steps that exists in current anti-customization methods, and gives detailed analysis regarding the perceptual discrepancy of diffusion models at different timesteps and intermediate layers, we have better aligned the optimization of adversarial gradients with diffusion models.
- Based on analysis, we propose a simple but effective anti-

customization method, including adaptive greedy time selection and a feature interference loss to improve the protection ability. Our method can be easily generalized to different anti-customization frameworks and improve their performance.
- Extensive evaluations on two face datasets demonstrate that our method achieves more obvious disruption of the user's identity and provides better privacy protection.

## 2. Related work

### 2.1. Generative Models and Diffusion Models

Variational autoencoders (VAEs) [21] and Generative adversarial networks (GANs) [8] are popular frameworks among generative models which have strong generative ability. These models encode the data $x$ as latent variables $z$ and model the joint distribution $p_\theta(x, z)$. However, the quality of VAE samples is not competitive with GANs which are suffering from training instability [10]. Since diffusion probabilistic models (DM) [31] progressively add noise to the data from the joint distribution $q_\theta(x_{0:T}|x_0)$ and denoise step by step, the efficiency of training and quality of samples have achieved state-of-work.

Unconditional generative models cannot produce desired samples and then models take different input as guidance have sprung up. Based on GAN, cGAN [25] generates images conditioned on the given labels $y$ and Cycle-GAN [38] implements unpaired image translation considering the given image. Equipped with some techniques like classifier-free guidance [11], the diffusion model gains the ability to follow diverse prompts as conditions during generation.

The open-sourced Latent Diffusion Models (LDMs) [27] operate images in the latent space of low dimensions rather than pixel space which greatly reduces training computation. To support different condition inputs, they add cross-attention layers into the underlying U-Net backbone as conditioning mechanisms $\tau_\theta$. The above delicate designs make diffusion more friendly for users to create what they need. It can also assist them in designing their diffusions which synthesize visual contents that contain specific concepts (*e.g.*, objects or styles) given by users during inference.

### 2.2. Customization

DreamBooth [28] stands out as a popular diffusion-based method for customizing text-to-image generation. This approach involves presenting $3 \sim 5$ images depicting a specific concept (*e.g.*, a particular dog) alongside a corresponding identifier (*e.g.*, "a sks dog"). DreamBooth utilizes these images to fine-tune the pre-trained Stable Diffusion model. This fine-tuning process encourages the model to "memorize" the concept and its identifier, enabling it to reproduce this concept in new contexts during inference.

On the other hand, Textual-inversion [7] employs a different approach. This method freezes the U-Net and exclusively optimizes the text embedding of unique identifiers (*e.g.* "sks") to represent the input concepts.

Inspired by DreamBooth, numerous works have been mushroomed such as custom-diffusion [22], Sine [37], among others. In the quest for more efficient fine-tuning, DreamBooth has a successful integration with LoRA [13] and has become a very influential customization project in the community. LoRA specifically decomposes the attention layer of the vision transformer model [6] into low-rank matrices reducing the cost associated with fine-tuning.

### 2.3. Privacy Protection for Diffusion Models

To alleviate the issue that private images are misused by Stable Diffusion based customization, so-called "anti-customization", some researchers [23, 30, 36] have recently delved into privacy protection for diffusion models. AdvDM [23] misleads the feature extraction of diffusion models. They analyze the training objective of fine-tuning and propose a direct way that uses the gradient of denoising loss as guidance to optimize the latent variables sampled from the denoising process. The generated adversarial examples degrade the generation ability of DreamBooth or other DM-based customized approaches. Inspired by [14], Anti-dreambooth [36] uses the alternating surrogate and perturbation learning (ASPL) to approximate the real trained models. They train the model on clean data and use these models as the surrogate model to compute the noise added to the user-provided images. The perturbed images then as the training data for surrogate model fine-tuning which mimics the real scenes. Photoguard [29] specially focuses on unauthorized image inpainting which misleads the public and does harm to personal reputation. They use both VAE encoder attack and UNet attack targeted on a gray image to hinder infringers from creating fake news.

Some concerns limit the application of current anti-customization methods in more practical scenarios, such as ineffective optimization, poor identity disruption, and simply using the reconstruction loss as guidance to generate adversarial examples [15–19]. Our paper aims to fill these blanks by figuring out the internal mechanisms that affects the performance of protection.

## 3. Method

### 3.1. Preliminaries

**Diffusion Model (DM)** DM is a probabilistic generative model that samples from Gaussian distribution and then progressive denoise to learn the desired data distribution. Given $x_0 \sim q(x)$, the forward process adds increasing noise to the input images at each time-step $t \in (0, T)$ which produces a sequence of noisy latent, $\{x_0, x_1, ..., x_T\}$. The backward process trains model $\epsilon_\theta(x_t, t, c)$ to predict the

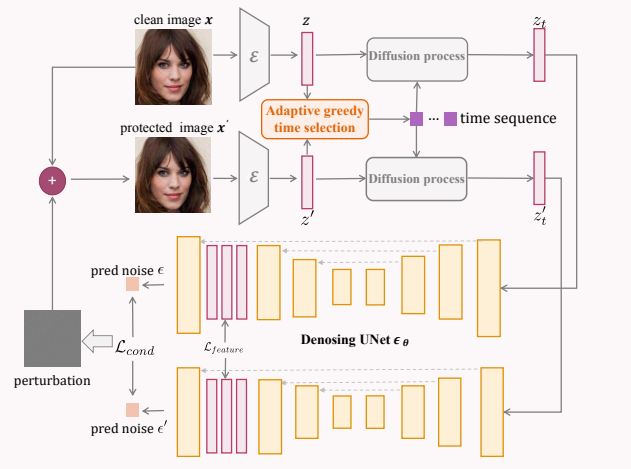



Figure 3. Pipeline of SimAC. We first greedily select the time step with our adaptive strategy during the feed forward phase. Then we integrate the feature interference loss with the vanilla training loss as the final objective. The noise is iteratively optimized.

noise added in $x_t$ to infer $x_{t-1}$. During denoising, the training loss is $l_2$ distance, as shown in Eq.(1), Eq.(2). Although there are many implementations of text-to-image diffusion models, Stable Diffusion is one of the few open-sourced diffusion models and is widely used in the community. Thus, our method is mainly based on the checkpoints of stable diffusion provided in HuggingFace.

$$\mathcal{L}_{uncond}(\theta, x_0) = \mathbb{E}_{x_0,t,\epsilon\in\mathcal{N}(0,1)}\|\epsilon - \epsilon_\theta(x_{t+1}, t)\|_2^2, \quad (1)$$

$$\mathcal{L}_{cond}(\theta, x_0) = \mathbb{E}_{x_0,t,c,\epsilon\in\mathcal{N}(0,1)}\|\epsilon - \epsilon_\theta(x_{t+1}, t, c)\|_2^2, \quad (2)$$

where $x$ is the input image, $t$ is the corresponding timestep, $c$ is condition input, (*e.g.*, text or image), $\epsilon$ is the noise term.

**Adversarial Attack** The adversarial examples for classification are crafted to mislead the model to classify the given input to the wrong labels. However, the traditional attack strategies are not effective when dealing with generative models. For diffusion models, adversarial examples are some images that are added on imperceptible noise, causing diffusion models to consider them out of the generated distribution. In detail, this noise heightens the challenge of image reconstruction and hinders the customization capabilities of applications based on diffusion models (DMs). The optimized noise is often typically constrained to be smaller than a constant value $\eta$. The $\delta$ is determined through the following formation:

$$\delta_{adv} = \arg\max_{\|\delta\|_p<\eta} L(f_\theta(x+\delta), y_{real}), \quad (3)$$

where $x$ is the input image, $y_{real}$ is the real images, and $L$ is the loss function used to evaluate the performance of adversarial examples.

Projected Gradient Descent (PGD) [24] is a widely used method to iteratively optimize adversarial examples. The process is formulated as

$$x^{t+1} = \prod_{(x,\eta)} \left(x^t + \alpha\,\mathrm{sgn}\left(\nabla_x L(f_\theta(x+\delta), y_{real})\right)\right) \quad (4)$$

where $x^0 = x$ and $x$ is the input image, $\mathrm{sgn}(\cdot)$ is sign function, $(\nabla_x L(f_\theta(x+\delta), y_{real})$ is the gradient of the loss function with respect to $x+\delta$. $\alpha$ represents the step size during each iteration and $t$ is the iteration number. With the operation $\Pi_{(x,\eta)}$, the noise is limited to a $\eta$-ball ensuring the adversarial examples are acceptable.

### 3.2. Overview

Here we delve into the properties of LDMs and analyze the potential vulnerabilities that can benefit the attack. In Sec. 3.3, we give a comprehensive analysis of the properties. In Sec. 3.4 and 3.5, we propose our method based on these analyses, mainly including two components, *i.e.*, adaptive greedy time interval selection and the feature interference loss. The overall pipeline is illustrated in Figure 3.

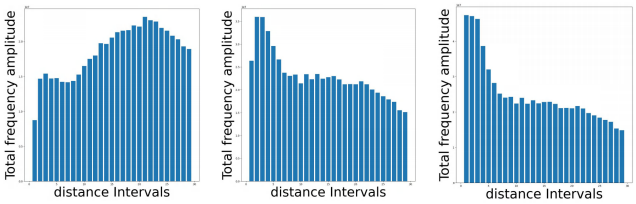
### 3.3. Analysis on Properties of LDMs

**Differences at different time step.** Our exploration focuses on analyzing gradients across various time intervals from a statistical standpoint. We've established a threshold of $1e^{-10}$ to assess the number of gradients with absolute values below this threshold in perturbed images across different timesteps. Figure 1 illustrates a notable trend: during the latter part of the total time interval $(0, MaxTrainStep)$, there's a sharp increase in the count of absolute gradients below the specified threshold. Additionally, Figure 1 presents the maximum value, mean, and median of absolute gradient decreases in the perturbed images when timesteps get larger.

As shown Eq.(5), the forward process admits sampling $x_t$ at an arbitrary timestep $t$. The magnitude of the noise corresponding to timestep is determined by a pre-defined noise schedule. To ensure that the final latent $x_T$ conforms to a standard normal distribution, the amplitude of noise injection gradually increases with the timestep.

$$q(\mathbf{x}_t \mid \mathbf{x}_0) = \mathcal{N}\left(\mathbf{x}_t; \sqrt{\bar{\alpha}_t}\mathbf{x}_0, (1-\bar{\alpha}_t)\,\mathbf{I}\right) \quad (5)$$

During denoising, we reconstruct the input image $x_0$ based on the noisy latent $z_t$ and visualize the difference between the reconstructed image and the input image in the frequency domain at different timesteps. As shown in Figure 4, when $t \in [0, 100]$, the main difference between the original and reconstructed images lies in the high-frequency components. Our initial expectation is to disrupt the reconstruction of the original image by introducing high-frequency adversarial noise, aiming to achieve anti-customization. How-




**t ∈ [0, 100]** **t ∈ [400, 500]** **t ∈ [700, 800]**

Figure 4. Frequency domain residual analysis. The X-axis is the distance from the low-frequency center (from low frequency to high frequency), and the Y-axis represents total magnitude of frequency residual. It can be seen that when $t$ is small, the high-frequency difference exceeds the low-frequency difference. As $t$ increasing, the low-frequency difference gradually dominated.

ever, when $t$ increases, such as $t \in [700, 800]$, the difference in low-frequency components between the reconstructed images and the input images dominates most of the frequency domain differences between the two. Therefore, this can explain why, when the timestep of noise scheduler is large, the gradient of the perturbed image is close to zero and result in ineffective noise optimization.

**Differences at Different UNet Layers.** Inspired by [35], we utilize PCA to visualize the output features of each layer in the U-Net decoder blocks during denoising. The decoder block consists of self-attention, cross-attention, and residual blocks, we select features of residual blocks. There are 11 layers in total, and the layers are visualized at timestep=500. As Figure 5 shows, with the output feature of UNet decoder blocks increasing, the visualized features gradually change from depicting structures and other low-frequency information to capturing texture and similar high-frequency information. Since our noise is intended to disrupt high-frequency information, the deeper features are good perturbation objects for our adversarial noises to reinforce interference which attempts to perturb high-frequency components in generated images.

### 3.4. Adaptive Greedy Time Interval Selection

To achieve more effective privacy protection within a limited number of noise injection steps, we propose a fast adaptive time interval selection strategy. Firstly, based on the input image, we randomly select five timesteps from the interval (0, max training step). For each time step, we compute the gradient concerning the input image. The absolute values of the gradients are summed. If the sum of absolute gradient values is the minimum at timestep $t$, the corresponding interval $(t-20, t+20)$ is then removed. For the timestep with maximal sum, noise is computed and added to the image and this noised image becomes the input for the next round of gradient computation. This process is repeated until the final time interval length is no greater than 100. We obtain the ultimate interval used for noise injection.

Our method relies on PGD[24] (Projected Gradient Descent), an iterative approach for adversarial attacks. At each iteration, computing the gradient becomes pivotal as it signifies the rate of change of the objective function concerning the perturbed images. The gradient approaching zero implies the possibility of iterative optimization but with minimal alterations to the objective function. This ultimately leads to reduced attack efficiency and diminished effectiveness. Thus, randomly selecting timesteps makes it more challenging to optimize the objective function under the same noise budget and results in decreased efficiency and effectiveness in countering customization. This highlights the crucial need for employing our adaptive, greedy time interval selection for perturbation.

**Algorithm 1:** Adaptive Greedy Time Interval Selection

**Input:** clean image $x$; model parameter $\theta$; number of search steps $N$; step length $\alpha$.
**Output:** time interval $seq$.

Initialize $x^{'} \leftarrow x, seq = (0, MaxTrainStep)$;
**for** $i = 1$ *to* $N$ **do**
  **if** *length of* $seq > 100$ **then**
    Sample timestep set $ts = (t_1, t_2, t_3, t_4, t_5)$ from $seq$,
    **for** $t$ *in* $ts$ **do**
      $sum(\nabla_{x_t^i} \mathcal{L}_{DM}(\theta))$
      **if** $sum$ *at* $t$ *is Minimum* **then**
        delete $(t-20, t+20)$ from $seq$
      **end**
      **if** $sum$ *at* $t$ *is Maximum* **then**
        $x^{'} = x + \alpha sgn((\nabla_{x_t^i} \mathcal{L}_{DM}(\theta))$
      **end**
    **end**
  **else**
    break
  **end**
**end**
**Result:** $seq$

**Analysis on efficacy.** Here we simply analyze our adaptive greedy selection for timestep $t$ (the above algorithm ). Its key steps in each iteration are randomly sampling 5 timesteps $T_s = \{t_j\}_{j=1}^{5}$ and rescaling the timestep range by deleting the interval of $t \in T_s$, which leads to the smallest gradient absolute value. Thus avoiding very small or even zero gradients during optimization.

**Theorem 1.** *Suppose the timestep range is rescaled from* $T = A \cup B$ *to* $T' = (A \setminus \Delta A) \cup (B \setminus \Delta B)$ *in some iteration, where timestep* $t \in B$ *leads to zero gradients, satisfying* $A \cap B = \varnothing$*, and* $\Delta A \cup \Delta B$ *denotes the deleted interval, then we have* $E_{t\sim T}|\nabla_x \mathcal{L}_{DM}| < E_{T_s \sim T} E_{t \sim T'} |\nabla_x \mathcal{L}_{DM}|$.

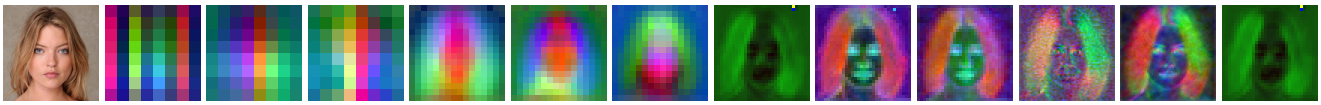



Figure 5. PCA visualization of features. The output feature of residual blocks are visualized at timestep=500. As feature goes deeper, the high-frequency information captured by the feature becomes more significant.


Figure 6. Diffusion feature over different timesteps. As the timestep of the noise scheduler increasing, part of the high-frequency information is lost and it is not good for adding high-frequency adversarial noise. This means that choosing an appropriate time sequence is also beneficial to feature interference loss.

*Proof.* We can easily obtain $E_{T_s\sim T}\frac{|\Delta A|}{|A|} < E_{T_s\sim T}\frac{|\Delta B|}{|B|}$ via proof by contradiction. For simplicity, we denote $\nabla_x\mathcal{L}_{DM}$ as $g(t)$, $LHS$ and $RHS$ denote left and right, then

$$\begin{aligned} LHS &= p(t\in A)E_{t\sim A}|g(t)| + p(t\in B)E_{t\sim B}|g(t)| \\ &= \frac{|A|}{|A|+|B|}E_{t\sim A}|g(t)| \approx E_{T_s\sim T}\frac{|A|\cdot E_{t\sim A\setminus\Delta A}|g(t)|}{|A|+|B|} \\ &< E_{T_s\sim T}\frac{|A|-|\Delta A|}{|A|+|B|-|\Delta A|-|\Delta B|}E_{t\sim A\setminus\Delta A}|g(t)| \\ &= E_{T_s\sim T}E_{t\sim A\setminus\Delta A}p(t\in A\setminus\Delta A)|g(t)| = RHS \end{aligned}$$

□

### 3.5. Feature Interference Loss

Based on the analysis in Sec 3.3, we propose the feature interference loss and integrate it with the vanilla training loss as the overall objective for optimization. This loss calculates the layer-wise euclidian distance between the intermediate features in some specially selected layers, *i.e.*,

$$\mathcal{L}_f = \mathbb{E}\|f_l^{t^*} - f_l^t\|_2^2 \tag{6}$$

$$\mathcal{L} = \mathcal{L}_{cond} + \lambda\mathcal{L}_f \tag{7}$$

Where $f_l^{t^*}$ is the output features of the selected layers sets $l$ at timestep $t$ for the input image, $f_l^t$ is the output features for current perturbed images, and $\lambda$ represents the weighting coefficient for the feature interference loss.

We visualized the output features of layer 7 at different time steps. As Figure 6 shows, when the timesteps increase, more noise is added to the latent and the response to the high-frequency components of the input image gradually decreases. Therefore, utilizing feature interference loss as the optimization objective for adversarial noise in small time intervals is a better choice to keep the effectiveness of adversarial noise disturbing high-frequency information.

## 4. Experiments

### 4.1. Setup

**Dataset** We utilized two facial datasets for experiments including Celeb-HQ [20] and VGGFace2 dataset[3]. The dataset comprises about 50 individuals and aligns with the Anti-DreamBooth, with each individual including at least 15 clear face images for customization. Since Anti-dreambooth needs alternating training, the dataset is divided into three sets, including set A, set B, and set C and each set contains 5 portrait images.

**Model** Since the open-sourced stable diffusion is the most popular implementation of latent diffusion among the community, our experiments mainly use the SD-v2.1 [1]. To test the performance of our method in a black-box scenario, we assume the versions of Stable Diffusion between anti-customization and customization are the same or different.

**Baseline** We compare several open-source methods that employ adversarial attacks on diffusion models to protect user images from being misused by text-to-image diffusion models, including Photoguard [29], AdvDM [23] and Anti-DreamBooth [36]. Due to the high GPU memory requirements of the complete PhotoGuard, we only utiliz the VAE encoder attack strategy in its paper for comparison.

**Metric** We utilize a face detector named Retinaface [5] to detect if there is a face in the image and use the Face Detection Failure Rate (FDFR) to assess the level of disruption to the generated faces. Upon detecting a face, we encode it using ArcFace [4] and calculate the cosine similarity between the protected image and clean input, measuring the identity resemblance between the detected face in the generated image and that of users. This matrix is defined as Identity Score Matching (ISM). In addition, the image quality is quantified by BRISQE [26], and the quality of detected facial images is measured through SER-FIQ [34].

**Implementation Details** We set the same noise budget for all methods $\eta = 16/255$. Additionally, the optimization steps and step size align with the settings specified in each baseline. The number of training epochs is 50 and each epoch includes 3 steps for surrogate model training and 9 steps for adversarial noise optimization. The step size for adding noise is set as 0.005 and the learning rate for the training model is set as 5e-7 which is appropriate for human faces. The maximum adaptive greedy search steps are set as 50 by default. After training, we used the noised im-

| CelebA-HQ | | | | |
|---|---|---|---|---|
| Method | "a photo of sks person" | | | |
| | ISM↓ | FDFR↑ | BRISQUE ↑ | SER-FQA↓ |
| PhotoGurad [29] | **0.25** | 41.09 | 19.38 | 0.55 |
| AdvDM [23] | 0.32 | 70.48 | 38.17 | 0.20 |
| Anti-DB [36] | 0.28 | 77.28 | 37.43 | **0.20** |
| Anti-DB + SimAC | 0.31 | **87.07** | **38.86** | 0.21 |
| Method | "a dslr portrait of sks person" | | | |
| | ISM↓ | FDFR↑ | BRISQUE ↑ | SER-FQA↓ |
| PhotoGurad [29] | 0.20 | 28.50 | 29.33 | 0.59 |
| AdvDM [23] | 0.25 | 65.37 | 37.86 | 0.41 |
| Anti-DB [36] | 0.19 | 86.80 | 38.90 | 0.27 |
| Anti-DB + SimAC | **0.12** | **96.87** | **42.10** | **0.15** |
| Method | "a photo of sks person looking at the mirror" | | | |
| | ISM↓ | FDFR↑ | BRISQUE ↑ | SER-FQA↓ |
| PhotoGurad[29] | 0.18 | 30.07 | 26.96 | 0.40 |
| AdvDM[23] | 0.29 | 35.10 | 36.46 | 0.36 |
| Anti-DB[36] | 0.22 | 42.86 | 40.34 | 0.28 |
| Anti-DB + SimAC | **0.12** | **91.90** | **43.97** | **0.06** |
| Method | "a photo of sks person in front of eiffel tower" | | | |
| | ISM ↓ | FDFR ↑ | BRISQUE↑ | SER-FQA↓ |
| PhotoGurad[29] | 0.08 | 50.95 | 32.82 | 0.40 |
| AdvDM[23] | 0.09 | 38.10 | 36.02 | 0.30 |
| Anti-DB[36] | 0.06 | 56.26 | 41.35 | 0.22 |
| Anti-DB + SimAC | **0.05** | **66.19** | **42.77** | **0.12** |

Table 1. Comparison with other open-sourced anti-customization methods on CelebA-HQ. We evaluate the performance under four different prompts during customization.

| VGG-Face2 | | | | |
|---|---|---|---|---|
| Method | "a photo of sks person" | | | |
| | ISM↓ | FDFR↑ | BRISQUE ↑ | SER-FIQ↓ |
| PhotoGurad [29] | **0.29** | 29.27 | 20.67 | 0.47 |
| AdvDM [23] | 0.32 | 63.07 | 38.51 | **0.21** |
| Anti-DB[36] | 0.30 | 64.67 | 37.89 | 0.22 |
| Anti-DB + SimAC | 0.31 | **80.27** | **40.71** | 0.22 |
| Method | "a dslr portrait of sks person" | | | |
| | ISM↓ | FDFR↑ | BRISQUE ↑ | SER-FIQ↓ |
| PhotoGurad [29] | 0.25 | 17.33 | 28.52 | 0.55 |
| AdvDM [23] | 0.30 | 67.80 | 37.58 | 0.35 |
| Anti-DB[36] | 0.23 | 77.47 | 38.79 | 0.29 |
| Anti-DB + SimAC | **0.11** | **96.33** | **41.78** | **0.12** |
| Method | "a photo of sks person looking at the mirror" | | | |
| | ISM↓ | FDFR↑ | BRISQUE ↑ | SER-FIQ↓ |
| PhotoGurad[29] | 0.17 | 40.87 | 30.10 | 0.32 |
| AdvDM[23] | 0.27 | 37.73 | 35.43 | 0.29 |
| Anti-DB[36] | 0.25 | 45.00 | 39.25 | 0.27 |
| Anti-DB + SimAC | **0.13** | **89.60** | **44.97** | **0.07** |
| Method | "a photo of sks person in front of eiffel tower" | | | |
| | ISM↓ | FDFR↑ | BRISQUE ↑ | SER-FIQ↓ |
| PhotoGurad[29] | 0.13 | 51.07 | 30.69 | 0.41 |
| AdvDM[23] | 0.14 | 38.67 | 35.99 | 0.31 |
| Anti-DB[36] | 0.10 | 54.93 | 41.13 | 0.23 |
| Anti-DB + SimAC | **0.09** | **61.20** | **42.17** | **0.10** |

Table 2. Comparison with other open-sourced anti-customization methods on VGG-Face2. We evaluate the performance under four different prompts during customization.

age for customization. The default base model is stable diffusion v2.1 combined with DreamBooth. According to the above analysis of the middle layer of the model, we select the 9, 10, and 11 layer features that can best represent the high-frequency information of the image for additional disturbance. The weight $\lambda$ of feature interference loss is set to 1. After finetuning 1000 steps, we save the model checkpoints and conduct inference. For each prompt, 30 images in .png format are generated for metric calculation.

## 4.2. Comparison with Baseline Methods

**Quantitative Results** To test the effectiveness of our approach in enhancing the protection of users' portrait images, we conduct a quantitative comparison under three prompts in Table 1 and Table 2. To be more practical, we list four text prompts for inference, the first one "a photo of sks person" is the same as the prompt used in training, while the other three were prompts unseen before. For each prompt, we randomly select 30 generated images to compute each metric, reporting their average values.

We can find that our method greatly improves the performance of Anti-DreamBooth and outperforms other baselines across all prompts. Due to our analysis of the unique properties of the diffusion model during denoising and effective addition of high-frequency noise, the face detection failure rates greatly increase, and the identity matching scores between the detected face and the input image are the lowest among all methods. This implies that our method is more effective in resisting abuse from customization.

**Qualitative Results** We show some results in Figure 7. It's evident that SimAC combined with Anti-DreamBooth achieves a strong image disruption effect, providing the best privacy protection for the input portrait. Since PhotoGuard operates an attack on the latent space, the resulting images tend to generate patterns similar to the target latent. This attack doesn't effectively reduce the probability of facial appearances or improve the quality of generated images.

Both AdvDM and Anti-DreamBooth use DM's training loss as an objective for optimizing noise, which, as been seen, yields similar results. The two methods aim to disrupt high-frequency components in images via adversarial noise to create artifacts and raise the bar for potential misuse of users' photos. Although the quality of generated images decreases, the results of both AdvDM and Anti-DreamBooth retain many details from the user input images. This indicates a leak of privacy for some users and isn't user-friendly. According to our analysis, both of these methods follow the training process of DM, randomly selecting timestamps for optimization. This leads to a waste of some noise addition steps, thereby reducing the efficiency of the attack and the effectiveness of protection.

## 4.3. Black-Box Performance

**Prompt Mismatch** When attackers use stable diffusion to customize concepts, the prompt they use might differ from our assumptions when adding noise. Thus, we use the

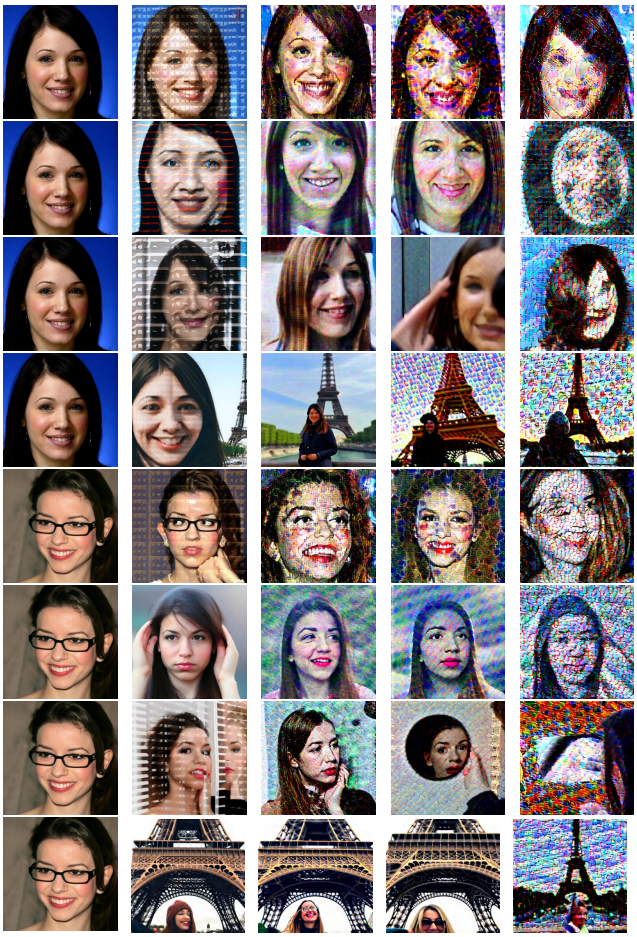



Figure 7. Visualization results (four prompts) on CelebA-HQ. From the first row to the last row is "a photo of sks person","a dslr portrait of sks person", "a photo of sks person looking at the mirror", "a photo of sks person in front of eiffel tower".

"a photo of sks person" during perturbation learning and change the rare identifier "sks" to "t@t" during DreamBooth model finetuing. The results in Table 3 indicate a decrease in performance in this scenario, but we notice that the Identity Score Matching(ISM), representing identity similarity remains consistent.

**Model Mismatch** The models used to add adversarial noise may also mismatch with the model fine-tuned by DreamBooth. We examine the effectiveness of adversarial noise learned on stable diffusion v2.1 against customization based on stable diffusion v2.1 and v1.4, or vice versa in Table 4. The conclusion of model mismatch resembles those of prompt mismatch. Although there is an overall decline in performance, the critical matrix, ISM, remains considerable. This also implies that in a model mismatch scenario, SimAC + Anti-DreamBooth can still protect user portrait privacy.

| Train [v] | Test [v] | "a photo of [v] person" | | | |
|---|---|---|---|---|---|
| | | ISM↓ | FDFR↑ | BRISQUE ↑ | SER-FIQ↓ |
| sks | sks | 0.31 | 87.07 | 38.86 | 0.21 |
| sks | t@t | 0.30 | 81.36 | 39.67 | 0.31 |
| Train [v] | Test [v] | "a dslr portrait of [v] person" | | | |
| | | ISM↓ | FDFR↑ | BRISQUE ↑ | SER-FIQ↓ |
| sks | sks | 0.12 | 96.87 | 42.10 | 0.15 |
| sks | t@t | 0.23 | 54.42 | 37.77 | 0.52 |
| Train [v] | Test [v] | "a photo of [v] person looking at the mirror" | | | |
| | | ISM↓ | FDFR↑ | BRISQUE ↑ | SER-FIQ↓ |
| sks | sks | 0.12 | 91.90 | 43.97 | 0.06 |
| sks | t@t | 0.16 | 30.54 | 40.63 | 0.25 |
| Train [v] | Test [v] | "a photo of [v] person in front of eiffel tower" | | | |
| | | ISM↓ | FDFR↑ | BRISQUE ↑ | SER-FIQ↓ |
| sks | sks | 0.05 | 66.19 | 42.77 | 0.12 |
| sks | st@t | 0.06 | 23.20 | 37.74 | 0.28 |

Table 3. Prompt mismatch between training and testing on CelebA-HQ. The training prompt is "a photo of sks person" and the inference prompt uses rare identifier "sks" or "t@t".

| Train | Test | "a photo of sks person" | | | |
|---|---|---|---|---|---|
| | | ISM↓ | FDFR↑ | BRISQUE ↑ | SER-FIQ↓ |
| v2.1 | v2.1 | 0.31 | 87.07 | 38.86 | 0.21 |
| v1.4 | v2.1 | 0.38 | 70.07 | 39.22 | 0.35 |
| v2.1 | v1.4 | 0.01 | 99.86 | 62.09 | 0.02 |
| Train | Test | "a dslr portrait of sks person" | | | |
| | | ISM↓ | FDFR↑ | BRISQUE ↑ | SER-FIQ↓ |
| v2.1 | v2.1 | 0.12 | 96.87 | 42.10 | 0.15 |
| v1.4 | v2.1 | 0.14 | 95.17 | 40.95 | 0.23 |
| v2.1 | v1.4 | 0.26 | 20.54 | 16.44 | 0.57 |
| Train | Test | "a photo of sks person looking at the mirror" | | | |
| | | ISM↓ | FDFR↑ | BRISQUE ↑ | SER-FIQ↓ |
| v2.1 | v2.1 | 0.12 | 91.90 | 43.97 | 0.06 |
| v1.4 | v2.1 | 0.16 | 79.39 | 42.52 | 0.13 |
| v2.1 | v1.4 | 0.20 | 82.72 | 40.12 | 0.16 |
| Train | Test | "a photo of sks person in front of eiffel tower" | | | |
| | | ISM↓ | FDFR↑ | BRISQUE ↑ | SER-FIQ↓ |
| v2.1 | v2.1 | 0.05 | 66.19 | 42.77 | 0.12 |
| v1.4 | v2.1 | 0.06 | 61.36 | 41.51 | 0.10 |
| v2.1 | v1.4 | 0.08 | 53.67 | 20.56 | 0.22 |

Table 4. Model version mismatch during training and testing on CelebA-HQ. The training prompt is "a photo of sks person"

## 5. Conclusion

Current anti-customization methods, primarily relying on adversarial attacks, often overlook crucial internal properties of the diffusion model, leading to ineffective optimization. This paper addresses this issue by exploring inherent properties to enhance anti-customization. Two key aspects are examined: Analyzing the relationship between timestep selection and the model's perception in the frequency domain inspires an adaptive greedy search for optimal timesteps. Scrutinizing feature roles during denoising, resulting in a sophisticated feature-based optimization framework. Experiments show increased identity disruption, enhancing user privacy and security.

## References


[1] Stable-diffusion-2-1. https://huggingface.co/stabilityai/stable-diffusion-2-1. 6

[2] Making pictures of trump getting arrested while waiting for trump's arrest. https://twitter.com/EliotHiggins/status/1637927681734987777. 1

[3] Qiong Cao, Li Shen, Weidi Xie, Omkar M Parkhi, and Andrew Zisserman. Vggface2: A dataset for recognising faces across pose and age. In *2018 13th IEEE international conference on automatic face & gesture recognition (FG 2018)*, pages 67–74. IEEE, 2018. 6

[4] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4690–4699, 2019. 6

[5] Jiankang Deng, Jia Guo, Evangelos Ververas, Irene Kotsia, and Stefanos Zafeiriou. Retinaface: Single-shot multi-level face localisation in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5203–5212, 2020. 6

[6] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 3

[7] 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. *arXiv preprint arXiv:2208.01618*, 2022. 1, 3

[8] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014. 3

[9] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations (ICLR)*, 2015. 2

[10] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. *Advances in neural information processing systems*, 30, 2017. 3

[11] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021. 3

[12] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 1

[13] 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. *arXiv preprint arXiv:2106.09685*, 2021. 1, 3

[14] Qidong Huang, Jie Zhang, Wenbo Zhou, Weiming Zhang, and Nenghai Yu. Initiative defense against facial manipulation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 1619–1627, 2021. 3

[15] Qidong Huang, Xiaoyi Dong, Dongdong Chen, Hang Zhou, Weiming Zhang, and Nenghai Yu. Shape-invariant 3d adversarial point clouds. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15335–15344, 2022. 3

[16] Qidong Huang, Xiaoyi Dong, Dongdong Chen, Hang Zhou, Weiming Zhang, Kui Zhang, Gang Hua, and Nenghai Yu. Pointcat: Contrastive adversarial training for robust point cloud recognition. *arXiv preprint arXiv:2209.07788*, 2022.

[17] Qidong Huang, Xiaoyi Dong, Dongdong Chen, Yinpeng Chen, Lu Yuan, Gang Hua, Weiming Zhang, and Nenghai Yu. Improving adversarial robustness of masked autoencoders via test-time frequency-domain prompting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1600–1610, 2023.

[18] Qidong Huang, Xiaoyi Dong, Dongdong Chen, Weiming Zhang, Feifei Wang, Gang Hua, and Nenghai Yu. Diversity-aware meta visual prompting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10878–10887, 2023.

[19] Qidong Huang, Xiaoyi Dong, Pan Zhang, Bin Wang, Conghui He, Jiaqi Wang, Dahua Lin, Weiming Zhang, and Nenghai Yu. Opera: Alleviating hallucination in multi-modal large language models via over-trust penalty and retrospection-allocation. *arXiv preprint arXiv:2311.17911*, 2023. 3

[20] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018. 6

[21] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 3

[22] Nupur Kumari, Bingliang Zhang, Richard Zhang, Eli Shechtman, and Jun-Yan Zhu. Multi-concept customization of text-to-image diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1931–1941, 2023. 1, 3

[23] 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 *International Conference on Machine Learning, ICML 2023*, pages 20763–20786. 2, 3, 6, 7

[24] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *6th International Conference on Learning Representations, ICLR 2018*. 4, 5

[25] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014. 3

[26] A. Mittal, A. K. Moorthy, and A. C. Bovik. No-reference image quality assessment in the spatial domain. *IEEE Transactions on Image Processing*, page 4695–4708, 2012. 6

[27] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image

synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 3

[28] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22500–22510, 2023. 1, 3

[29] Hadi Salman, Alaa Khaddaj, Guillaume Leclerc, Andrew Ilyas, and Aleksander Madry. Raising the cost of malicious ai-powered image editing. In *International Conference on Machine Learning, ICML 2023*, pages 29894–29918. 3, 6, 7

[30] 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. *arXiv preprint arXiv:2302.04222*, 2023. 3

[31] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pages 2256–2265. PMLR, 2015. 1, 3

[32] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *9th International Conference on Learning Representations, ICLR*, 2021. 1

[33] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In *International Conference on Learning Representations (ICLR)*, 2013. 2

[34] Philipp Terhörst, Jan Niklas Kolf, Naser Damer, Florian Kirchbuchner, and Arjan Kuijper. SER-FIQ: unsupervised estimation of face image quality based on stochastic embedding robustness. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 5650–5659. IEEE, 2020. 6

[35] Narek Tumanyan, Michal Geyer, Shai Bagon, and Tali Dekel. 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*, pages 1921–1930, 2023. 5

[36] 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 *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2116–2127, 2023. 2, 3, 6, 7

[37] Zhixing Zhang, Ligong Han, Arnab Ghosh, Dimitris N Metaxas, and Jian Ren. Sine: Single image editing with text-to-image diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6027–6037, 2023. 3

[38] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017. 3