# PID: Prompt-Independent Data Protection Against Latent Diffusion Models

**Ang Li** [1] **Yichuan Mo** [2] **Mingjie Li** [3] **Yisen Wang** [2 4]

## Abstract

The few-shot fine-tuning of Latent Diffusion Models (LDMs) has enabled them to grasp new concepts from a limited number of images. However, given the vast amount of personal images accessible online, this capability raises critical concerns about civil privacy. While several previous defense methods have been developed to prevent such misuse of LDMs, they typically assume that the textual prompts used by data protectors exactly match those employed by data exploiters. In this paper, we first empirically demonstrate that breaking this assumption, i.e., in cases where discrepancies exist between the textual conditions used by protectors and exploiters, could substantially reduces the effectiveness of these defenses. Furthermore, considering the visual encoder's independence from textual prompts, we delve into the visual encoder and thoroughly investigate how manipulating the visual encoder affects the few-shot fine-tuning process of LDMs. Drawing on these insights, we propose a simple yet effective method called **Prompt-Independent Defense (PID)** to safeguard privacy against LDMs. We show that PID can act as a strong privacy shield on its own while requiring significantly less computational power. We believe our studies, along with the comprehensive understanding and new defense method, provide a notable advance toward reliable data protection against LDMs. Our code is available at https://github.com/PKU-ML/Diffusion-PID-Protection

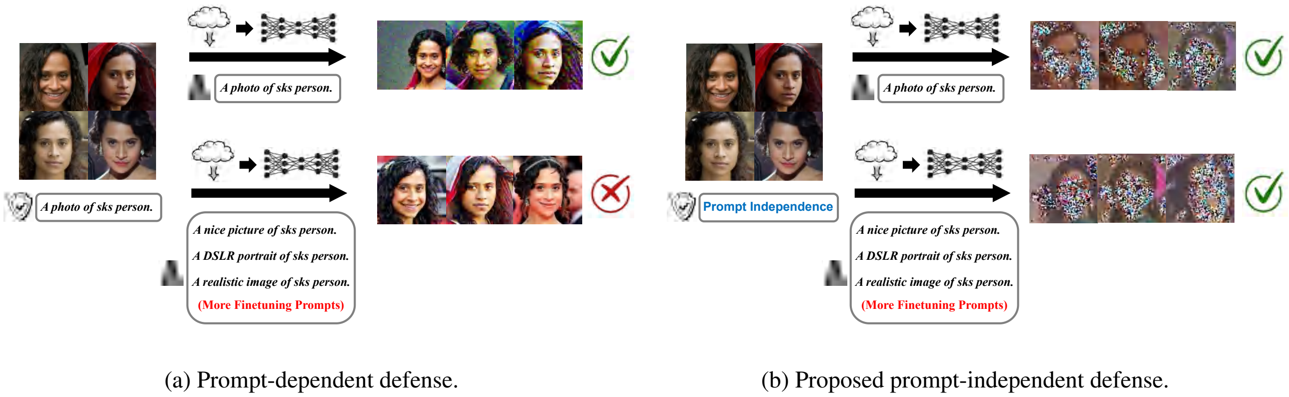
## 1. Introduction

The advent of Latent Diffusion Models (LDMs) has ushered in an era where images of unprecedented quality are synthesized, blurring the lines between artificial creations and authentic human-generated content, including portraits, photographic arts, and drawings (Song & Ermon, 2019; Song et al., 2020; Rombach et al., 2022; Ramesh et al., 2022; Holz, 2022; Podell et al., 2024). A particularly intriguing aspect of LDMs is the capability of few-shot fine-tuning, a.k.a. personalization of the generative model, which teaches the models a brand new concept, such as human faces or painting styles, with as few as 4∼5 images in a matter of minutes (Hu et al., 2021; Ruiz et al., 2023; Gal et al., 2023; Clark et al., 2024). However, the ease with which targeted sets of images can be curated with either manual downloading or web crawling on social media renders this capability a double-edged sword. Several selfies casually posted online could mean an array of counterfeit images produced by the LDM fine-tuned by malicious users with the photos, showing exactly the same person clothless or in places he/she has never been to. Civilians are concerned by lawsuits and news related to the unregulated exploitation of such techniques (Juefei-Xu et al., 2022). Thus, developing reliable data protection algorithms that prevent the malicious misuse of LDMs on unauthorized images is vital for both the research community and society.

Fortunately, notable efforts have been made to protect images containing sensitive information like human faces or unique art styles against such exploitations by generative models (Yeh et al., 2020; Ruiz et al., 2020; Huang et al., 2021a;b; Wang et al., 2022a; Yang et al., 2021; Li et al., 2023b). Salman et al. (2023) protect images from malicious image editing. Liang et al. (2023) adopt the adversarial attack (Goodfellow et al., 2014; Wang et al., 2019; 2022b; Mo et al., 2022; Li et al., 2023a) against LDMs to hinder the models from learning features from the protected data. Van Le et al. (2023) propose to generate protective perturbations by attacking a fully trained surrogate model or by synchronizedly disrupting the training process. More recently, Xue et al. (2023) propose Score Distillation Sampling (SDS) to lighten the computational overhead required by optimizing the protective perturbations. These works have made significant steps toward the ultimate goal. However, when it comes to few-shot fine-tuning, previous works assume

[1] School of EECS, Peking University, China [2] National Key Lab of General Artificial Intelligence, School of Intelligence Science and Technology, Peking University, China [3] CISPA Helmholtz Center for Information Security, Germany [4] Institute for Artificial Intelligence, Peking University, China. Correspondence to: Yisen Wang <yisen.wang@pku.edu.cn>.

*Proceedings of the 41st International Conference on Machine Learning*, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).


A photo of sks person.
A photo of sks person.
A nice picture of sks person.
A DSLR portrait of sks person.
A realistic image of sks person.
(More Finetuning Prompts)
Prompt Independence
A photo of sks person.
A nice picture of sks person.
A DSLR portrait of sks person.
A realistic image of sks person.
(More Finetuning Prompts)

(a) Prompt-dependent defense.

(b) Proposed prompt-independent defense.

*Figure 1.* Influence of the prompt misalignment, i.e., $c_{prot} \neq c_{explo}$, on the performance of the prompt-related protection (Figure 1a) and the prompt-independent one (Figure 1b). In each sub-figure, the left-most component depicts the data protection stage and whether a textual prompt is involved. The middle component exhibits the data exploiters collect the protected images and try to fine-tune a latent diffusion model with matched/mismatched prompts. The right-most component displays some generated images by the generative models fine-tuned with different prompts. The images are all generated with *A high-quality portrait of sks person.* The instance is from the CelebA-HQ dataset (Liu et al., 2015) and the fine-tuned model is Stable Diffusion v1.5 (Rombach et al., 2022).

to a large degree that the data protection stage (where we add protective perturbations to the images) and the data exploitation stage (where malicious fine-tuning happens), are conditioned on the *identical* textual prompts. Since the data protectors have no prior knowledge of the exploiters, the assumption on prompt consistency may not be realistic in practice.

The visual encoder in LDMs projects high-resolution images into a condensed latent space where the diffusion process takes place. Despite its important role, the component has been an *overlooked* part of the protection. Previous studies make arbitrary choices for the visual encoder, like manipulating the mean value of the latent representations (Liang & Wu, 2023; Salman et al., 2023). Note that the independence of the visual encoder from the textual prompts allows it to be unaffected by the assumption of prompt consistency, intuitively making it the right fit for strengthening the protections against varied prompts. Therefore, we raise the following Research Questions (**RQs**).

- **RQ1:** Does a mismatch between the prompts used in the protection and exploitation stages affect the efficacy of existing defense algorithms?
- **RQ2:** How do perturbations in pixel space affect the output of the visual encoders in LDMs and thus affect the fine-tuning process?
- **RQ3:** If the answer to RQ1 is **yes**, can we improve the robustness of the protection by making better use of the prompt-independent visual encoders?

In this paper, we first investigate the robustness of current defense approaches under the prompt-mismatch scenario. To simulate an adversarial environment where the exploiters intentionally craft textual prompts to undermine the defense, we define a set of candidate prompts, denoted as $c_{prot}$, for the exploiters to choose when fine-tuning the latent diffusion models (the full list can be found in Appendix A). We randomly draw an individual from the CelebA-HQ (Liu et al., 2015) dataset and protect the images of it with a typical algorithm ASPL (Van Le et al., 2023) using its recommended hyper-parameters. During the protection stage, we fix the textual prompt to be *a photo of sks person* (denoted as $c_{prot}$). We then separately fine-tune the Stable Diffusion v1.5 with DreamBooth (Ruiz et al., 2023) conditioned on each of the malicious candidate prompts ($c_{explo}$). Lastly, we generate images using the fine-tuned models with the prompt *a high-quality portrait of sks person* and show some of the generated images in Figure 1a. For the case where $c_{prot} \neq c_{explo}$, the displayed images are drawn from the visually optimal model among the candidate models. We observe the protective performance of the prompt-dependent defense is notably weakened by the intentionally varied prompts. We hypothesize that the degradation is caused by the entanglement between the perturbations and the textual condition. Deeply concerned by the above observations, we delve into the latent space in LDMs and fully investigate the possibility of utilizing the visual encoder to construct data protection that is more robust to varied prompts. Based upon our findings, we propose a new defense family featuring **Prompt-Independent Defense (PID)**. PID is completely independent of textual prompts, showing robustness to varied fine-tuning prompts as we show qualitatively in Figure 1b, and quantitively in Section 6. Our main contributions are summarized as follows:

- We empirically observe that mismatched prompts between the protection stage and the exploitation stage could undermine the effectiveness of current data protection algorithms.
- We thoroughly explore the possibility of leveraging the visual encoder within LDMs for more robust data protection and propose a new algorithm named PID.
- Through extensive validation, we demonstrate the efficacy of PID against different training algorithms, datasets, and adaptive attacks.

## 2. Related Work

### 2.1. Diffusion Models

**Diffusion models** are a type of generative models (Sohl-Dickstein et al., 2015; Ho et al., 2020) that learns the data distribution via two opposing procedures: a forward pass and a backward pass. Given an input image $\boldsymbol{x_0} \sim q(x)$, the forward pass gradually adds noise to the image following a noise scheduler $\{\beta_t : \beta_t \in (0,1)\}_{t=1}^{T}$ until the data approximately becomes Gaussian noise. For each timestep $t$, the perturbed image is given by $\boldsymbol{x_t} = \sqrt{\tilde{\alpha}_t}\boldsymbol{x_0} + \sqrt{1-\tilde{\alpha}_t}\boldsymbol{\varepsilon}$, where $\alpha_t = 1 - \beta_t, \tilde{\alpha}_t = \Pi_{s=1}^{t}\alpha_s$ and $\boldsymbol{\varepsilon} \sim \mathcal{N}(0, \mathbf{I})$. The reverse process is to reconstruct $\boldsymbol{x_0}$ from $\boldsymbol{x_T}$ via step-by-step predicting the noise added. Specifically, the noise $\boldsymbol{\varepsilon}$ at timestep $t$ is estimated by a parameterized network $\boldsymbol{\varepsilon_\theta}(\boldsymbol{x_{t+1}}, t)$. The training loss is commonly defined as the $\ell_2$ distance between the actual noise and the prediction

$$L_{unc}(\theta, \boldsymbol{x_0}) = \mathbb{E}_{\boldsymbol{x_0}, t, \boldsymbol{\varepsilon} \in \mathcal{N}(0,\mathbf{I})} ||\boldsymbol{\varepsilon} - \boldsymbol{\varepsilon_\theta}(\boldsymbol{x_{t+1}}, t)||_2^2, \quad (1)$$

where $t$ is uniformly sampled from $\{1, 2, \cdots, T\}$ and *unc* stands for unconditional diffusion model.

**Text-to-Image Latent Diffusion Models** get rid of the massive computational cost brought by operations in pixel space via projecting images into the latent space defined by a pre-trained image encoder (Radford et al., 2021; Kingma & Welling, 2013), of which the most widely used implementation is the KL-based VAE (Thomas & Cover, 1991) as adopted by (Rombach et al., 2022; Podell et al., 2024; Peebles & Xie, 2023). In this work, we primarily focus on the KL-based VAE while we note that the idea of using the visual encoder for prompt-independent defense is not restricted by the concrete implementation. Denoting the KL-based VAE as $\mathcal{E}$, the latent distribution of data $\boldsymbol{x}$ is given by $\mathcal{N}(\mu_{\mathcal{E}}(\boldsymbol{x}), \sigma^2_{\mathcal{E}}(\boldsymbol{x})) := \mathcal{E}(\boldsymbol{x})$, and the latent representation of data $\boldsymbol{x}$ is sampled from the distribution via reparametrization, $\boldsymbol{z} = \mu_{\mathcal{E}}(\boldsymbol{x}) + \sigma_{\mathcal{E}}(\boldsymbol{x})\varepsilon := \mathcal{E}(\boldsymbol{x}, \varepsilon)$, where $\varepsilon \in \mathcal{N}(0, \mathbf{I})$. Furthermore, the textual condition $c$ is involved in utilizing the cross-attention (Vaswani et al., 2017; Balaji et al., 2022; Nichol et al., 2022; Rombach et al., 2022; Saharia et al., 2022) between the UNet (Ronneberger et al., 2015) and an extra text encoder (Radford et al., 2021). With the condition $c$ and the latent representation $\boldsymbol{z_0} = \mathcal{E}(\boldsymbol{x_0}, \varepsilon)$, the training process is re-formulated as follows

$$L_{cond}(\theta, c, \boldsymbol{z_0}) = \mathbb{E}_{\boldsymbol{z_0}, t, \boldsymbol{\varepsilon}} ||\boldsymbol{\varepsilon} - \boldsymbol{\varepsilon_\theta}(\boldsymbol{z_{t+1}}, c, t)||_2^2. \quad (2)$$

**Personalization of LDMs** (Hu et al., 2021; Gal et al., 2023; Ruiz et al., 2023; Clark et al., 2024) enables users to fine-tune the LDMs with only a handful of images. After fine-tuning, the LDMs usually exhibit an astonishing grasp of the concepts contained in the images and can flexibly combine the new concepts with the original training data, synthesizing images that never existed before. The technique seems to be a double-edged sword that raises the potential threat to civil privacy and artists' copyrights to a degree that cannot be ignored anymore.

### 2.2. Data Protection against LDM

Worried by the malicious use of LDMs, a series of works have made significant contributions to defend personal images against LDMs. There are two main threads of current works: 1) generating adversarial examples with a surrogate model, specifically AdvDM (Liang et al., 2023), Mist (Liang & Wu, 2023), photoguard (Salman et al., 2023), and FSGM (Van Le et al., 2023); and 2) generating unlearnable examples (Huang et al., 2021a; Ren et al., 2022) with a bilevel optimization (ASPL (Van Le et al., 2023)). Concretely, the former type of defense first fine-tunes a surrogate model $\theta_{sur}$ with the clean data. Then it adversarially maximizes the training loss of $\theta_{sur}$ on the perturbed data:

$$\theta_{sur} = \arg\min_{\theta} L_{cond}(\theta, c, \mathcal{E}(\boldsymbol{x})), \quad (3)$$

$$\boldsymbol{x^*} = \underset{||\boldsymbol{x'} - \boldsymbol{x}||_p \le \boldsymbol{\varepsilon}}{\arg\max} \ L_{cond}(\theta_{sur}, c, \mathcal{E}(\boldsymbol{x'})), \quad (4)$$

where $\boldsymbol{x^*}$ denotes the adversarial examples, i.e., the protected images, and $\boldsymbol{\varepsilon}$ ensures the invisibility of the perturbation. Similar to the idea of classic Unlearnable Examples (Huang et al., 2021a), the latter form of defense proposes to generate the protected images alongside the training procedure with a min-max optimization:

$$\boldsymbol{x^*} = \underset{||\boldsymbol{x'} - \boldsymbol{x}||_p \le \boldsymbol{\varepsilon}}{\arg\max} \ \underset{\theta}{\arg\min} L_{cond}(\theta, c, \mathcal{E}(\boldsymbol{x'})). \quad (5)$$

It is important to note that both of the above algorithms require **a textual prompt** $c$ to protect the images, which makes the perturbations inherently correlated with the text condition. Besides, back-propagating through the large UNet costs enormous GPU VRAM (around 24GB without extra tricks). There also exist protection algorithms involving the visual encoders, either by manipulating the mean value of the latent distribution targetedly (Liang & Wu, 2023; Salman et al., 2023), or by directly making the latent representations unrelated to the data (Liang et al., 2023). However, previous works have not fully explored the potential of visual encoders, and our empirical studies in Section 4 render the above-mentioned choices suboptimal.

## 3. Is Prompt-related Defense Robust to Varied Prompts?

In this section, we conduct quantitive evaluations on the robustness of prompt-related defense when confronted with the changed prompts in a realistic setting.

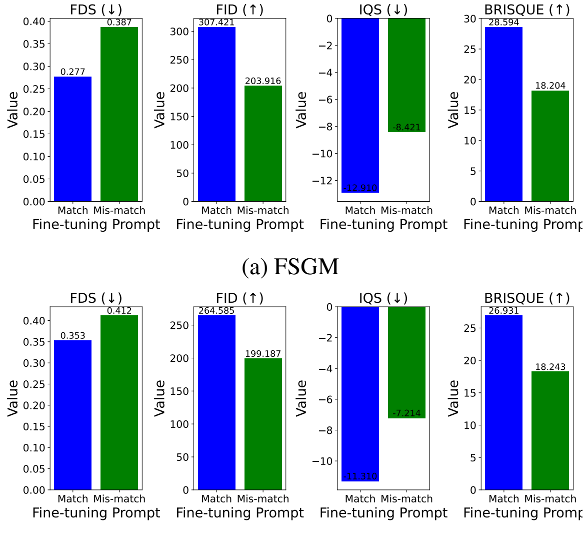
We begin by introducing our overall experimental setup while details are listed in Appendix A for brevity.

**Data & Model:** Our experiments primarily utilize the CelebA-HQ (Liu et al., 2015) dataset where we randomly select 10 celebrities and choose 4 images for each. Following Van Le et al. (2023), we use Stable Diffusion v1.5 (Rombach et al., 2022) as the default model and DreamBooth (Ruiz et al., 2023) as the default fine-tuning method.

**Defense:** We consider the method of FSGM and ASPL from Van Le et al. (2023), whose objectives are entirely correlated with the textual prompts. The perturbation budget is set to 0.05 and the perturbed images are saved in PNG format in this paper unless otherwise specified.

**Metrics:** We use two metrics to measure the similarity between the generated images and the training images: Face Detection Score (FDS) (Zhang et al., 2016) and Fréchet Inception Distance (FID) (Heusel et al., 2017). Additionally, we use two metrics to assess image quality: Image Quality Score (IQS) (Radford et al., 2021) and Blind/Referenceless Image Spatial Quality Evaluator (BRISQUE) (Mittal et al., 2012). We define ↑ (value increasing) and ↓ (value decreasing) to indicate the direction of better protection effect, e.g., a larger FID indicates a greater distance between the distribution of the generated images and the training images, suggesting that the generated images do not capture the training data well, thus protecting the privacy of the training data.

**Results:** For the selected 4 images of each celebrity, we adopt the defense method FSGM and ASPL with the protecting prompt $\boldsymbol{c_{prot}}$ to generate the corresponding protected version. These protected images are then used to fine-tune the model with the fine-tuning prompt $\boldsymbol{c_{explo}}$, resulting in different fine-tuned models[1]. For testing, we use arbitrary prompts to generate a set of images, which are then evaluated using the four metrics mentioned above. The average results across different fine-tuned models are shown in Figure 2. We can see that the protection performance is notably affected when the protecting prompt differs from the fine-tuning prompt. For example on the FSGM method, when the fine-tuning prompts do not match the protecting prompts, the metric FDS increases over 35% ($0.277 \rightarrow 0.387$) and the metric FID decreases 30% ($307.421 \rightarrow 203.916$). The phenomenon is consistent for other metrics and methods.



(a) FSGM

(b) ASPL

*Figure 2.* The quantitative results showing the performance of the prompt-related defenses when textual prompts between the protection stage and the exploration stage are matched ($\boldsymbol{c_{prot}} = \boldsymbol{c_{explo}}$) and mismatched ($\boldsymbol{c_{prot}} \neq \boldsymbol{c_{explo}}$).

Deeply concerned by our observation that breaking the prompt-consistency assumption made by the data protectors could enable the exploiters to generate high-quality mimic images, even when the data is safeguarded to some extent, we aim to design a prompt-agnostic defense in the following parts.

## 4. Does Perturbing the Visual Encoer Affect Fine-tuning?

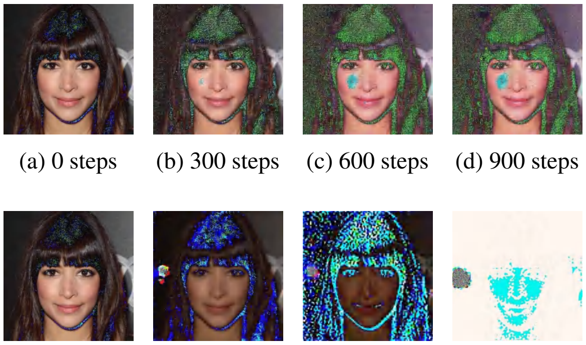
Recall that the latent distribution is modeled by a KL-based VAE (Kingma & Welling, 2013) as a multinomial Gaussian Distribution, $\mathcal{N}(\mu_{\mathcal{E}}(\boldsymbol{x}), \sigma^2_{\mathcal{E}}(\boldsymbol{x}))$, which is prompt-independent. This property can be leveraged to address the defense degradation when there is a prompt mismatch. Before delving into this potential solution, we first investigate how changes in the latent distribution, i.e., in the mean $\mu_{\mathcal{E}}(\boldsymbol{x})$ and the variance $\sigma^2_{\mathcal{E}}(\boldsymbol{x})$, influence fine-tuning.

We define $L_{mean}$ to maximize the distance between the mean of the perturbed images and the clean images, while $L_{var}$ to maximize the distance between the variances of the

[1] When $\boldsymbol{c_{prot}} \neq \boldsymbol{c_{explo}}$, we choose the fine-tuned model that has the highest FDS.

*Table 1.* Evaluation of fine-tuning on the images which maximize the mean distance $L_{mean}$ and the variance distance $L_{var}$ respectively. *Random* denotes adding random noise uniformly within $[-\varepsilon, \varepsilon]$ to the clean image.

| Data | FDS (↓) | FID (↑) | IQS (↓) | BRISQUE (↑) |
|---|---|---|---|---|
| Clean | 0.480 | 144.570 | 4.310 | 15.447 |
| Random | 0.479 | 150.788 | 4.504 | 12.160 |
| $L_{mean}$ | 0.370 | 243.292 | **-5.373** | **21.655** |
| $L_{var}$ | **0.329** | **265.337** | -0.926 | 16.369 |



(a) 0 steps (b) 300 steps (c) 600 steps (d) 900 steps

(e) 0 steps (f) 300 steps (g) 600 steps (h) 900 steps

*Figure 3.* Visualizations of the perturbed latent representations. We decode the latent representations $z$ obtained during the maximization of $L_{mean}$ and $L_{var}$ with the visual decoder in the LDM. (a) to (d) corresponding to the change of mean, while (e) to (h) for the variance. The images are obtained with the Stable Diffusion v1.5.

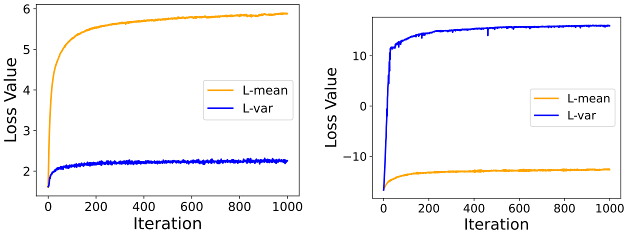
two distributions. Formally, $L_{mean}$ and $L_{var}$ are

$$L_{mean}(\boldsymbol{x}, \boldsymbol{\delta}) = ||\mu_{\mathcal{E}}(\boldsymbol{x} + \boldsymbol{\delta}) - \mu_{\mathcal{E}}(\boldsymbol{x})||_2^2, \quad (6)$$

$$L_{var}(\boldsymbol{x}, \boldsymbol{\delta}) = ||\sigma_{\mathcal{E}}(\boldsymbol{x} + \boldsymbol{\delta}) - \sigma_{\mathcal{E}}(\boldsymbol{x})||_2^2, \quad (7)$$

where $\boldsymbol{\delta}$ denotes the perturbation added. We maximize the above loss functions with $\ell_\infty$-PGD$_{1000}$ (Madry et al., 2018) and $\boldsymbol{\delta}$ is constrained by $||\boldsymbol{\delta}||_\infty \leq \varepsilon = 0.05$. Then we conduct fine-tuning on the images obtained by optimizing the above two targets and evaluate the fine-tuned models with the same evaluation framework in Section 3.

The results presented in Table 1 demonstrate that significantly reshaping the latent distribution indeed has a substantial influence on fine-tuning. To visually illustrate the influence of the distorted latent distribution, we decode the representations $\boldsymbol{z}$ sampled from the distribution during the optimization process using the visual decoder and display the decoded images in Figure 3[2]. Combing results in Table 1 and Figure 3, we find that a large mean difference with the clean images mainly influences the texture of the output images, making them appear covered with heavy noise (low IQS and high BRISQUE). Conversely, a large variance significantly prohibits the model from grasping the core concepts of the images (low FDS and high FID).



(a) $||\mu_{\mathcal{E}}(x + \delta) - \mu_{\mathcal{E}}(x)||_2^2$ (b) $\log \sigma_{\mathcal{E}}(x + \delta)^2 - \log \sigma_{\mathcal{E}}(x)^2$

*Figure 4.* The change of the latent distribution as the perturbations are added. (a) the change of the $\ell_2$ distance between the mean of the clean and the perturbed latent distribution. (b) the change of the $\ell_2$ distance between the variance of the clean and the perturbed latent distribution. The target of each colored line is shown in the figure legend.

Lastly, we plot the $\ell_2$ norm of the mean difference and the variance difference in Figure 4, which reveals that even small perturbations added in the pixel space (0.05) can significantly alter the latent distribution. The variance changes so drastically that the gap between the variance of clean images and that of perturbed images ranges from approximately $\sim 2^{-15}$ to $\sim 2^{12}$. Additionally, we observe that changes in the mean and changes in the variance are not entirely correlated. Whatever in Figures 4a or 4b, one undergoes significant fluctuations while the other does not exhibit substantial variation, which indicates their distinct impacts on fine-tuning results.

Overall, by inducing perturbations in the pixel space, we can manipulate the two statistics of the latent distribution, thereby significantly affecting different aspects of the fine-tuning outcome.

## 5. How Can We Make Better Use of the Visual Encoder for Data Protection?

As discussed in Section 4, perturbing the latent distribution significantly impacts the fine-tuning process, and notably, this latent distribution is prompt-independent. Therefore, in this section, we aim to utilize the visual encoder to implement an effective prompt-independent defense mechanism.

### 5.1. Proposed Prompt-Independent Defense (PID)

From the results in Table 1, we know that influencing the mean and variance impacts different aspects of the learning procedure. Observing Figure 4, we know that altering just one of these statistics is insufficient to simultaneously in-

[2] Note that, for this specific experiment, we directly input $(\boldsymbol{x} + \boldsymbol{\delta})$ as float numbers to the encoder without converting to uint-8. This approach maximally showcases the perturbations' influence.

duce substantial changes in both. This observation triggers us to explore the possibility of manipulating the latent distribution more effectively by designing a sophisticated target that leverages the benefits of influencing both the mean and the variance.

We first attempt to disrupt the representations $\boldsymbol{z} = \mathcal{E}(\boldsymbol{x}, \boldsymbol{\varepsilon})$ sampled from the latent distribution, leading to the loss function $L_{sample}$, which is also employed by Liang et al. (2023). To reduce unnecessary randomness in the optimization process, we then experiment with excluding $\boldsymbol{\varepsilon}$ from $L_{sample}$, resulting in the loss function $L_{add}$. Considering the significant disparity in the magnitudes of the mean and variance ($10^2$ vs $10^{-3}$) observed in Figure 4, we propose $L_{add-log}$, which jointly optimizes the logarithm of the variance and the mean. Additionally, we explore targeted manipulation of the mean $\boldsymbol{x_{target}}$, as done by Liang & Wu (2023), where the target is the default image specified in their paper[3]. We denote this loss as $L^T_{mean}$.

$$L_{sample}(\boldsymbol{x}, \boldsymbol{\delta}) = \mathbb{E}_{\boldsymbol{\varepsilon_1}, \boldsymbol{\varepsilon_2}} ||\mathcal{E}(\boldsymbol{x} + \boldsymbol{\delta}, \boldsymbol{\varepsilon_1}) - \mathcal{E}(\boldsymbol{x}, \boldsymbol{\varepsilon_2})||_2^2, \tag{8}$$

$$L_{add}(\boldsymbol{x}, \boldsymbol{\delta}) = L_{mean} + L_{var}, \tag{9}$$

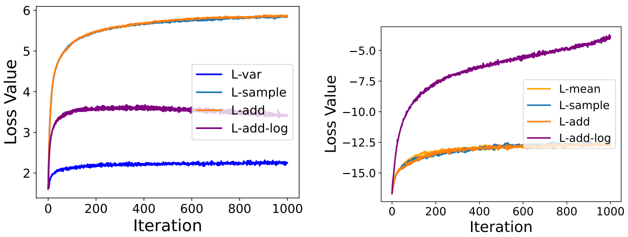
$$L_{add-log}(\boldsymbol{x}, \boldsymbol{\delta}) = L_{mean} + \log \frac{\sigma_{\mathcal{E}}(\boldsymbol{x} + \boldsymbol{\delta})^2}{\sigma_{\mathcal{E}}(\boldsymbol{x})^2}, \tag{10}$$

$$L^T_{mean}(\boldsymbol{x}, \boldsymbol{\delta}) = -||\mu_{\mathcal{E}}(\boldsymbol{x} + \boldsymbol{\delta}) - \mu_{\mathcal{E}}(\boldsymbol{x_{target}})||_2^2, \tag{11}$$

We proceed by evaluating the influence of the defense targets proposed above on the latent distributions, as done in Figure 4. Notably, we notice that $L_{add-log}$ (the purple line in Figure 5a and Figure 5b) is the only defense target that shifts both statistics away from their normal values significantly with averaged $\ell_2$ distance of the mean being 3.5 and 0.06 for variance. On the contrary, $L_{sample}$ and $L_{add}$ perform significantly worse in perturbing variance.

Equipped with a suitable target, we next examine whether it has a larger influence on fine-tuning than before. The results presented in Table 2 reveal that the potential of the encoders in data protection has not been fully explored before. The loss functions adopted by previous literature, $L_{mean}$, $L^T_{mean}$, and $L_{add}$, exhibit sub-optimal performance compared to $L_{add-log}$. We note that the similar behaviors of $L_{mean}$, $L_{sample}$, and $L_{add}$ can be well explained by observations in Figure 5a and Figure 5b, as all of them mostly focus on the mean value.

A carefully designed optimization target, $L_{add-log}$, proves to combine the advantages of influencing $\mu$ and $\sigma$. Not only does it successfully stop the model from learning the human face (low FDS, high FID), but it also notably affects the structure and texture of the output images (low IQS and high BRISQUE). We are surprised to find that $L_{add-log}$ even outperforms both $FSGM(c = c_1)$ and $ASPL(c = c_1)$ under this fine-tuning configuration. Given that the latter defenses require much more GPU memory since they involve the UNet (Ronneberger et al., 2015) model, which is much heavier than the visual encoder, we thus believe the visual encoder should play an undiminished role in protecting data against LDMs. For its superior protection effect and independence of textual conditions, we implement the defense target defined by $L_{add-log}$ as the **Prompt-Independent Defense (PID)**.



(a) $||\mu_{\mathcal{E}}(x + \delta) - \mu_{\mathcal{E}}(x)||_2^2$ (b) $\log \sigma_{\mathcal{E}}(x + \delta)^2 - \log \sigma_{\mathcal{E}}(x)^2$

*Figure 5.* The change of the latent distribution as the perturbations are added. $L_{add-log}$ is the only loss that has a significant impact on both statistics. (a) and (b) We plot the change of the mean and the variance of perturbed images as the maximization of each loss goes respectively. The results are averaged over all elements in the tensor.

*Table 2.* Evaluation of fine-tuning on the images maximizing the losses defined from Equation (8) to (11), together with $L^T_{mean}$. The fine-tuned model is Stable Diffusion v1.5.

| Data | FDS (↓) | FID (↑) | IQS (↓) | BRISQUE (↑) |
|---|---|---|---|---|
| Clean | 0.480 | 144.570 | 4.310 | 15.447 |
| Random | 0.479 | 150.788 | 4.504 | 12.160 |
| $L^T_{mean}$ | 0.377 | 271.540 | -4.047 | 28.622 |
| $L_{sample}$ | 0.377 | 265.588 | -5.135 | 21.119 |
| $L_{add}$ | 0.377 | 268.260 | -5.500 | 20.465 |
| $L_{add-log}$ | **0.329** | **411.990** | **-18.296** | **35.510** |

## 5.2. Integrating PID with Existing Defenses

We continue to explore the possibility of improving the current defenses with PID. To combine the two distinct types of defense, namely defense with the encoder and defense with attacking the training loss function, we adopt a joint optimization approach involving a weighted combination of both two defense objectives, similar to Liang et al. (2023) and Liang & Wu (2023). Specifically, given a defense target $T$ that incorporates the training loss of LDMs and a defense target aimed at manipulating the latent distribution, $L$, we define a tradeoff coefficient $\lambda$ to balance the two targets. The combined defense is expressed as follows:

$$L_{combo}(\theta, c, \boldsymbol{x}) = T(\theta, c, \boldsymbol{x}) + \lambda L(\boldsymbol{x}). \tag{12}$$

[3] https://github.com/mist-project/mist/blob/main/resources

Here, $\theta$ again denotes the model parameter, $c$ represents the textual condition, and $\boldsymbol{x}$ is the data to protect. $T(\theta, c, \boldsymbol{x})$ can be the defense methods defined in equation 3 and equation 5.

We let $L = L_{add-log}$ the strongest defense target we observed, and $T \in \{\text{ASPL, FSGM}\}$. We empirically identify $\lambda = 0.05$ as the best parameter in our default setting. While it's possible to exhaustively search for the optimal $\lambda$ across all settings, we consistently adopt $\lambda^* = 0.05$ in the following experiments due to the massive computational demands.

## 6. Experiments

In this part, we first assess the performance of PID as well as existing algorithms under both the $\boldsymbol{c_{prot}} = \boldsymbol{c_{explo}}$ and the $\boldsymbol{c_{prot}} \neq \boldsymbol{c_{explo}}$ scenarios. Second, we experiment with combining PID and existing defenses. Last but not least, we test the robustness of PID under harsh conditions.

### 6.1. PID Excels Regardless of the Prompt Consistency Assumption

**Experiment Setup:** We compare PID with the three symbolic defense methods, AdvDM (Liang et al., 2023), FSGM, and ASPL (Van Le et al., 2023) on the CelebA-HQ (Liu et al., 2015) and VGGFACE (Cao et al., 2018) dataset. We largely adopt their default configurations when running the defense algorithms. We generate PID with PGD$_{1000}$ (Madry et al., 2018) and the perturbation budget is set to $\varepsilon_\infty = 0.05$. We use the Stable Diffusion v1.5 and Stable Diffusion v2.1 (Rombach et al., 2022) as the base model[4]. The evaluation protocol is identical to the one introduced in Section 3 with the experimental details provided in Appendix A.

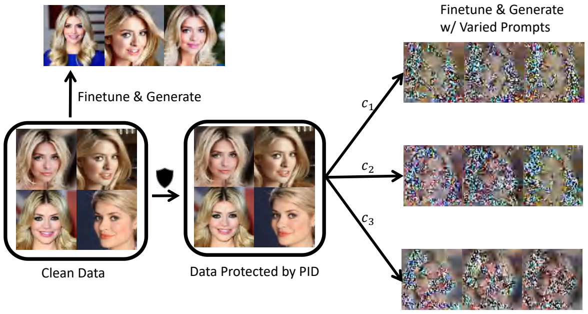


*Figure 6.* Illustration of PID's defense performance against varied prompts used by the data exploitors. $\boldsymbol{c_1}$ to $\boldsymbol{c_3}$ correspond to three different fine-tuning prompts. In the figure, we finetune the SD v1.5. More visualizations can be found in Appendix F.

**Results:** The complete results for CelebA-HQ are listed in Table 3. Remarkably, despite consuming significantly less computational resources (approximately 20% GPU memory, 5G v.s. 24G), PID achieves comparable, if not superior, performance compared to the three algorithms incorporating UNet across all four training configurations. Specifically, when the text encoder is frozen, i.e., not trained, during finetuning, PID consistently prohibits the LDMs from learning useful semantical information, resulting in notably poor facial similarity (0.254 for SD v1.5 and 0.285 for SD v2.1). Regarding the case where the text encoders are also finetuned, PID induces severely noisy, low-quality images that have little semantical correlation with the training data, as evidenced by the degraded FDS (0.303 and 0.288), substantially reduced IQS (-8.979 and -14.764), and high BRISQUE (28.927 and 50.112). Benefiting from the visual encoders' independence from the text encoders, PID consistently results in FID greater than 300 across all settings, rendering the generated images unrelated to the training data. The results demonstrate the potential for PID as a strong baseline method for safeguarding data against LDMs regardless of the varied prompts. The results for VGGFACE (Cao et al., 2018) and LoRA finetuning (Hu et al., 2021) are provided in Appendix C and we showcase images generated by the LDMs fine-tuned on the protected data in Appendix F.

### 6.2. Hybriding PID Robustifies Current Algorithms

We then compare the prompt-dependent defenses with their PID-hybridized variants. Results in Table 4 reveal that PID is able to enhance the robustness of the current algorithms. we observe that ASPL+PID is much more robust than ASPL regardless of whether the text encoder is frozen or not, as supported by the notably lower FDS (0.254 v.s. 0.370, 0.335 v.s. 0.412) and higher FID (352 v.s. 271, 208 v.s. 199). Moreover, FSGM+PID consistently results in images of worse semantic information than FSGM (lower FDS). Based on the above results, we argue that PID can be incorporated into existing defenses for more reliable data protection against LDMs.

We do not ignore the fact that combining PID with FSGM fails to do better in image quality, which might be attributed to a sub-optimal $\lambda^*$ choice or the difficulty of joint optimization. Though inferior, FSGM+PID still qualifies for a valid defense for its larger influence on semantic information (lower FDS).

### 6.3. Improved Cross-model Transferability

Since the data protectors have no control over what model the downstream exploiter uses, it is possible that different models are adopted in the two stages. We examine the transferability between different models of PID as well as existing algorithms under the $\boldsymbol{c_{prot}} \neq \boldsymbol{c_{explo}}$ setting.

[4] We obtain the Stable Diffusion v1.5 from https://huggingface.co/runwayml/stable-diffusion-v1-5 and the Stable Diffusion v2.1 from https://huggingface.co/stabilityai/stable-diffusion-2-1.

*Table 3.* Evaluation of defense algorithms under a controlled scenario $\boldsymbol{c_{prot}} = \boldsymbol{c_{explo}}$ and a more realistic scenario where $\boldsymbol{c_{prot}} \neq \boldsymbol{c_{explo}}$ on the CelebA-HQ dataset. The best-performing defense under each metric is marked with **bold**. Frozen text-encoder means we freeze the parameters of the text-encoder during fine-tuning, while unfrozen means we simultaneously fine-tune the text-encoder and UNet.

| Scenario | Data | Frozen Text Encoder | | | | Unfrozen Text Encoder | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | FDS(↓) | FID(↑) | IQS(↓) | BRISQUE(↑) | FDS(↓) | FID(↑) | IQS(↓) | BRISQUE(↑) |
| | Clean | 0.482 | 144.570 | 4.397 | 14.757 | 0.557 | 152.870 | 7.104 | 18.445 |
| $c_{prot} = c_{explo}$ | AdvDM | 0.344 | 240.452 | -11.310 | 19.100 | 0.358 | 208.859 | -8.558 | 25.472 |
| | FSGM | 0.342 | 246.434 | -8.710 | 22.046 | 0.277 | 307.421 | -12.910 | 28.594 |
| | ASPL | 0.330 | 295.415 | -9.558 | 26.993 | 0.353 | 264.585 | -11.310 | 26.931 |
| | PID | **0.205** | **411.990** | **-18.296** | **49.178** | **0.257** | **325.962** | **-29.693** | **62.749** |
| $c_{prot} \neq c_{explo}$ | AdvDM | 0.378 | 253.501 | -8.534 | 16.692 | 0.407 | 216.154 | **-9.260** | 21.813 |
| | FSGM | 0.374 | 224.460 | -5.607 | 20.678 | 0.387 | 203.916 | -8.421 | 18.204 |
| | ASPL | 0.370 | 185.074 | -3.669 | 26.993 | 0.412 | 199.187 | -7.214 | 18.243 |
| | PID | **0.254** | **352.795** | **-15.273** | **35.510** | **0.303** | **307.760** | -8.979 | **28.927** |

(a) The fine-tuned model is Stable Diffusion v1.5.

| Scenario | Data | Frozen Text Encoder | | | | Unfrozen Text Encoder | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | FDS(↓) | FID(↑) | IQS(↓) | BRISQUE(↑) | FDS(↓) | FID(↑) | IQS(↓) | BRISQUE(↑) |
| | Clean | 0.494 | 175.856 | 9.959 | 11.654 | 0.565 | 140.735 | 5.641 | 10.199 |
| $c_{prot} = c_{explo}$ | AdvDM | 0.322 | 252.407 | -2.127 | 24.382 | 0.362 | 229.829 | -12.287 | 31.833 |
| | FSGM | 0.298 | 277.588 | 0.586 | 30.837 | 0.312 | 257.165 | -7.456 | 30.764 |
| | ASPL | 0.300 | 282.938 | 0.012 | 31.429 | 0.313 | **266.097** | -5.707 | 28.832 |
| | PID | **0.255** | **350.382** | **-16.556** | **50.757** | **0.288** | 260.496 | **-14.764** | **50.112** |
| $c_{prot} \neq c_{explo}$ | AdvDM | 0.346 | 245.780 | -5.081 | 24.293 | 0.398 | 231.861 | -10.128 | 31.006 |
| | FSGM | 0.326 | 252.407 | 2.365 | 30.950 | 0.347 | 234.313 | -6.279 | 24.918 |
| | ASPL | 0.341 | 236.257 | -1.872 | 30.717 | 0.388 | 203.413 | -1.541 | 23.357 |
| | PID | **0.285** | **336.617** | **-12.634** | **43.746** | **0.288** | **366.596** | **-14.764** | **50.112** |

(b) The fine-tuned model is Stable Diffusion v2.1.

Specifically, we consider the transferability between the Stable Diffusion v1.5 and Stable Diffusion v2.1.

**Results:** PID enjoys great transferability between the two model versions as shown in Table 5, which might be due to the similarity in the condensed representations of images. Also, we notice that the transferability of existing algorithms from SD v2.1 to SD v1.5 is relatively weaker than the other way around.

### 6.4. Resilliance to Adaptive Attacks

We continue studying the robustness of PID when faced with adaptive attacks with our quantitative results reported in Table 6.

**Adaptive Attack:** Since our proposed PID focuses on manipulating the mean and the variance of the latent distribution, there could be adaptive attacks trying to break the conditions for our defense to be effective. We propose three possible adaptive attacks and test the robustness of our proposed defenses against them.

**(1) Zero $\sigma$:** As it is shown in Figure 5b, PID causes the variance of the latent distribution to increase dramatically. Therefore, the attack might fix the standard value of the latent distribution $\sigma_{\mathcal{E}}(\boldsymbol{x})$ of the perturbed images to be 0 to mitigate such effect. However, a zero standard value will make the finetuning process easier to overfit and lead to inferior generation results. Our results also reveal that PID works very well in such training settings with FDS$= 0.253$ and IQS$= -9.313$.

**(2) Clipped $\sigma$ & (3) Fixed $\sigma$:** A smarter attacker might try clipping or fixing the standard value $\sigma_{\mathcal{E}}(x)$ to a relatively normal value, e.g. $10^{-7}$, rather than directly fixing it to be $0$. Adopting the attack, we observe PID's influence on image quality is weakened, with the improved IQS and decreased FID shown in Table 6. However, the FDS is still very low (¡ 0.3), rendering the attack ineffective.

In all, PID is believed to be tolerant of the adaptive attacks we proposed above and exhibit convincing robustness.

### 6.5. Robustness against Data Corruptions

After releasing the protected data, the protectors have no control over what data exploiters will do to the images. Here we consider four common data corruptions that may influence the effect of the protective perturbations, namely **randomly resizing and cropping**, **smoothing with uni-**

*Table 4.* Evaluation of PID-hybridized defense algorithms under an uncontrolled scenario where $c_{prot} \neq c_{explo}$. The default training prompt *A photo of sks person* is adopted as $c_{prot}$. The backbone model is Stable Diffusion v1.5 and $\lambda^* = 0.05$. The superior one between FSGM/ASPL and FSGM+PID/ASPL + PID under each metric is marked with **bold**.

| Data | Frozen Text Encoder | | | | Unfrozen Text Encoder | | | |
|---|---|---|---|---|---|---|---|---|
| | FDS (↓) | FID (↑) | IQS (↓) | BRISQUE (↑) | FDS (↓) | FID (↑) | IQS (↓) | BRISQUE (↑) |
| FSGM | 0.374 | **224.460** | **-5.607** | 20.678 | 0.387 | **203.916** | **-8.421** | 18.204 |
| FSGM + $\lambda^*$PID | **0.276** | 185.704 | -3.669 | **21.096** | **0.303** | 185.074 | -3.665 | **19.096** |
| ASPL | 0.370 | 271.893 | -5.786 | 22.724 | 0.412 | 199.187 | **-7.214** | 18.243 |
| ASPL + $\lambda^*$PID | **0.254** | **352.795** | **-15.723** | **35.510** | **0.335** | **208.859** | -3.443 | **20.659** |

*Table 5.* The transferability of different data protection algorithms. The images are protected with the *Source* models and are exploited by the *Target* models. v1.5 and v2.1 denote Stable Diffusion v1.5 and Stable Diffusion v2.1 repectively. The text encoders are frozen when fine-tuning the models in this table.

| Src.→Dst. | Data | FDS (↓) | FID (↑) | IQS (↓) | BRISQUE (↑) |
|---|---|---|---|---|---|
| v2.1 → v1.5 | AdvDM | 0.371 | 223.914 | -2.083 | 30.010 |
| | FSGM | 0.364 | 208.278 | -2.339 | 37.563 |
| | ASPL | 0.311 | 252.740 | -2.510 | 37.706 |
| | PID | **0.268** | **350.069** | **-14.802** | **46.204** |
| v1.5 → v2.1 | AdvDM | 0.407 | 231.139 | -4.660 | 17.108 |
| | FSGM | 0.372 | 241.951 | -7.724 | 23.091 |
| | ASPL | 0.397 | 239.222 | -6.606 | 23.950 |
| | PID | **0.265** | **251.253** | **-15.087** | **24.365** |

*Table 6.* Robustness of PID against three adaptive attacks we proposed. All evaluations are done via fine-tuning a Stable Diffusion v1.5. The best-performing attack is marked as **bold**.

| Freeze-TE | Data | FDS (↓) | FID (↑) | IQS (↓) | BRISQUE (↑) |
|---|---|---|---|---|---|
| ✓ | Clean | 0.480 | 144.570 | 4.130 | 14.757 |
| | Zero $\sigma$ | **0.253** | **201.951** | -9.313 | 22.686 |
| | Clipped $\sigma$ | 0.249 | 207.611 | -12.968 | 33.156 |
| | Fixed $\sigma$ | 0.239 | 207.611 | **-6.238** | **22.685** |
| ✗ | Clean | 0.557 | 128.870 | 7.104 | 18.445 |
| | Zero $\sigma$ | 0.257 | 228.788 | **-7.793** | 36.637 |
| | Clipped $\sigma$ | **0.279** | 367.174 | -16.602 | 37.565 |
| | Fixed $\sigma$ | 0.249 | **207.260** | -13.215 | **30.168** |

**form noise**, **image denoising** [5] and **JPEG compression**. The model we used for this part is SD v1.5 and we freeze the text-encoder during fine-tuning. The reported experiments are done with $c_{prot} = c_{explo}$.

**Result:** Based on Table 7 we can observe that PID, the simplest defense among the four algorithms, withstands all four corruptions as evidenced by consistently low FDS and high FID. PID shows comparable performance to the AdvDM and FSGM even in its worst case, the JPEG compression. However, the huge performance drop when compressed still signals the need to design more robust protection algorithms against image compression.

*Table 7.* The robustness of the defensive algorithms against four commonly seen data corruptions. The model finetuned is SD v1.5 and the text-encoder is frozen. We assume $c_{prot} = c_{explo}$ for this specific experiment.

| Corruption | Data | FDS (↓) | FID (↑) | IQS (↓) | BRISQUE (↑) |
|---|---|---|---|---|---|
| Cropping | AdvDM | 0.379 | 258.085 | -1.460 | 22.655 |
| | FSGM | 0.376 | 255.739 | -0.573 | 18.926 |
| | ASPL | 0.369 | 268.892 | -1.850 | 22.319 |
| | PID | **0.246** | **275.468** | **-6.290** | **24.183** |
| Smoothing | AdvDM | 0.388 | 211.059 | **-3.351** | 18.013 |
| | FSGM | 0.388 | **229.721** | 1.391 | 16.403 |
| | ASPL | 0.377 | 223.193 | -2.210 | 17.212 |
| | PID | **0.213** | 184.483 | 0.108 | **47.121** |
| Denoising | AdvDM | 0.391 | 230.016 | **-1.656** | 20.457 |
| | FSGM | 0.396 | 230.049 | 2.108 | 17.639 |
| | ASPL | 0.372 | **248.910** | 0.326 | 21.292 |
| | PID | **0.213** | 184.483 | 0.108 | **42.440** |
| Compression | AdvDM | 0.386 | 229.973 | -5.768 | 25.340 |
| | FSGM | 0.390 | 225.208 | -3.547 | 24.042 |
| | ASPL | 0.354 | **267.039** | **-6.644** | **27.983** |
| | PID | **0.345** | 221.601 | 0.287 | 20.510 |

## 7. Conclusion

In this paper, we delve into the reliability of current data protection algorithms against LDMs without the prompt-consistency assumption. Our investigation reveals that the prompt-related defenses could suffer notable performance decreases when the data exploiters intentionally craft fine-tuning prompts. Motivated by the visual encoder's independence from the textual prompts, we thoroughly analyze how perturbing the visual encoder impacts the fine-tuning process and propose a prompt-independent defense algorithm named PID. With the empirically validated effectiveness of PID and its ability to enhance existing algorithms, we believe that our proposed prompt-independent algorithm marks an important step toward reliable protection of data from exploitation by LDMs.

## Acknowledgements

Yisen Wang was supported by National Key R&D Program of China (2022ZD0160300), National Natural Science Foundation of China (62376010, 92370129), Beijing Nova Program (20230484344), and CCF-Baichuan-EB Fund.

[5] We adopt the Gaussian image denoiser from the Aydin library, https://github.com/royerlab/aydin

## Impact Statement

Marching towards the overarching goal of privacy-conscious artificial intelligence, particularly in the realm of privacy-aware deep image generative models, our research offers significant empirical and technical advancements to this critical domain. With our work known by the broader civilians, we are confident that our work will bolster public trust in artificial intelligence, further mitigating the risks posed to personal privacy by generative models, and ultimately building trustworthy AI.

## References

Aydemir, A. E., Temizel, A., and Temizel, T. T. The effects of jpeg and jpeg2000 compression on attacks using adversarial examples. *arXiv preprint arXiv:1803.10418*, 2018.

Balaji, Y., Nah, S., Huang, X., Vahdat, A., Song, J., Kreis, K., Aittala, M., Aila, T., Laine, S., Catanzaro, B., et al. ediffi: Text-to-image diffusion models with an ensemble of expert denoisers. *arXiv*, 2022.

Cao, Q., Shen, L., Xie, W., Parkhi, O. M., and Zisserman, A. Vggface2: A dataset for recognising faces across pose and age. In *FG*, 2018.

Clark, K., Vicol, P., Swersky, K., and Fleet, D. J. Directly fine-tuning diffusion models on differentiable rewards. *ICLR*, 2024.

Gal, R., Alaluf, Y., Atzmon, Y., Patashnik, O., Bermano, A. H., Chechik, G., and Cohen-or, D. An image is worth one word: Personalizing text-to-image generation using textual inversion. In *ICLR*, 2023.

Goodfellow, I. J., Shlens, J., and Szegedy, C. Explaining and harnessing adversarial examples. 2014.

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., and Hochreiter, S. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *NeurIPS*, 2017.

Ho, J., Jain, A., and Abbeel, P. Denoising diffusion probabilistic models. *NeurIPS*, 2020.

Holz, D. MS Windows NT kernel description. `https://www.midjourney.com`, 2022.

Hu, E. J., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W., et al. Lora: Low-rank adaptation of large language models. In *ICLR*, 2021.

Huang, H., Ma, X., Erfani, S. M., Bailey, J., and Wang, Y. Unlearnable examples: Making personal data unexploitable. In *ICLR*, 2021a.

Huang, Q., Zhang, J., Zhou, W., Zhang, W., and Yu, N. Initiative defense against facial manipulation. *AAAI*, 2021b.

Juefei-Xu, F., Wang, R., Huang, Y., Guo, Q., Ma, L., and Liu, Y. Countering malicious deepfakes: Survey, battleground, and horizon. *IJCV*, 2022.

Kawar, B., Zada, S., Lang, O., Tov, O., Chang, H., Dekel, T., Mosseri, I., and Irani, M. Imagic: Text-based real image editing with diffusion models. In *CVPR*, 2023.

Kingma, D. P. and Welling, M. Auto-encoding variational bayes. *arXiv*, 2013.

Li, A., Wang, Y., Guo, Y., and Wang, Y. Adversarial examples are not real features. In *NeurIPS*, 2023a.

Li, Z., Yu, N., Salem, A., Backes, M., Fritz, M., and Zhang, Y. {UnGANable}: Defending against {GAN-based} face manipulation. In *USENIX*, 2023b.

Liang, C. and Wu, X. Mist: Towards improved adversarial examples for diffusion models. *arXiv*, 2023.

Liang, C., Wu, X., Hua, Y., Zhang, J., Xue, Y., Song, T., Zhengui, X., Ma, R., and Guan, H. Adversarial example does good: Preventing painting imitation from diffusion models via adversarial examples. In *ICML*, 2023.

Liu, Z., Luo, P., Wang, X., and Tang, X. Deep learning face attributes in the wild. In *ICCV*, 2015.

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. Towards deep learning models resistant to adversarial attacks. In *ICLR*, 2018.

Meng, C., He, Y., Song, Y., Song, J., Wu, J., Zhu, J.-Y., and Ermon, S. Sdedit: Guided image synthesis and editing with stochastic differential equations. *arXiv*, 2021.

Mittal, A., Moorthy, A. K., and Bovik, A. C. No-reference image quality assessment in the spatial domain. *IEEE Transactions on Image Processing*, 2012.

Mo, Y., Wu, D., Wang, Y., Guo, Y., and Wang, Y. When adversarial training meets vision transformers: Recipes from training to architecture. In *NeurIPS*, 2022.

Nichol, A. Q., Dhariwal, P., Ramesh, A., Shyam, P., Mishkin, P., Mcgrew, B., Sutskever, I., and Chen, M. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. In *ICLR*, 2022.

Peebles, W. and Xie, S. Scalable diffusion models with transformers. In *CVPR*, 2023.

Podell, D., English, Z., Lacey, K., Blattmann, A., Dockhorn, T., Müller, J., Penna, J., and Rombach, R. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *ICLR*, 2024.

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

Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., and Chen, M. Hierarchical text-conditional image generation with clip latents. *arXiv*, 2022.

Ren, J., Xu, H., Wan, Y., Ma, X., Sun, L., and Tang, J. Transferable unlearnable examples. *arXiv*, 2022.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022.

Ronneberger, O., Fischer, P., and Brox, T. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015.

Ruiz, N., Bargal, S. A., and Sclaroff, S. Disrupting deepfakes: Adversarial attacks against conditional image translation networks and facial manipulation systems. In *ECCVW*, 2020.

Ruiz, N., Li, Y., Jampani, V., Pritch, Y., Rubinstein, M., and Aberman, K. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *CVPR*, 2023.

Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E. L., Ghasemipour, K., Gontijo Lopes, R., Karagol Ayan, B., Salimans, T., et al. Photorealistic text-to-image diffusion models with deep language understanding. *NeurIPS*, 2022.

Salman, H., Khaddaj, A., Leclerc, G., Ilyas, A., and Madry, A. Raising the cost of malicious ai-powered image editing. In *ICML*, 2023.

Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., and Ganguli, S. Deep unsupervised learning using nonequilibrium thermodynamics. In *ICML*, 2015.

Song, Y. and Ermon, S. Generative modeling by estimating gradients of the data distribution. *NeurIPS*, 2019.

Song, Y., Sohl-Dickstein, J., Kingma, D. P., Kumar, A., Ermon, S., and Poole, B. Score-based generative modeling through stochastic differential equations. *ICLR*, 2020.

Thomas, J. A. and Cover, T. M. *Elements of Information Theory*. New York, 1991.

Van Le, T., Phung, H., Nguyen, T. H., Dao, Q., Tran, N. N., and Tran, A. Anti-dreambooth: Protecting users from personalized text-to-image synthesis. In *ICCV*, 2023.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. *NeurIPS*, 2017.

Wang, R., Huang, Z., Chen, Z., Liu, L., Chen, J., and Wang, L. Anti-forgery: Towards a stealthy and robust deepfake disruption attack via adversarial perceptual-aware perturbations. *IJCAI*, 2022a.

Wang, Y., Zou, D., Yi, J., Bailey, J., Ma, X., and Gu, Q. Improving adversarial robustness requires revisiting misclassified examples. In *ICLR*, 2019.

Wang, Y., Wang, Y., Yang, J., and Lin, Z. A unified contrastive energy-based model for understanding the generative ability of adversarial training. In *ICLR*, 2022b.

Wang, Z., Bovik, A., Sheikh, H., and Simoncelli, E. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 2004.

Xue, H., Liang, C., Wu, X., and Chen, Y. Toward effective protection against diffusion-based mimicry through score distillation. In *ICLR*, 2023.

Yang, C., Ding, L., Chen, Y., and Li, H. Defending against gan-based deepfake attacks via transformation-aware adversarial faces. In *IJCNN*, 2021.

Yeh, C.-Y., Chen, H.-W., Tsai, S.-L., and Wang, S.-D. Disrupting image-translation-based deepfake algorithms with adversarial attacks. In *WACVW*, 2020.

Zhang, K., Zhang, Z., Li, Z., and Qiao, Y. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Processing Letters*, 2016.

Zhang, R., Isola, P., Efros, A. A., Shechtman, E., and Wang, O. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018.

# A. Experimental Details

## A.1. Metric Definition

Throughout our paper we evaluate the fine-tuning results using four matrics, covering different properties of generated images. We define the *Face Detection Score (FDS)* as the average cosine similarity between the embeddings of training images and the embeddings of generated images, where the embedding are given by a MTCNN (Zhang et al., 2016) pre-trained on a large-scale facial dataset VGGFace2 (Cao et al., 2018). FDS mainly captures the semantic similarity between the images and evaluates whether the facial information is learned. Fréchet Inception Distance (FID) (Heusel et al., 2017) is a metric evaluating the distance between the distribution of the generated images and the distribution of training images, evaluating the model's master of the image from another perspective. The *Image Quality Score (IQS)* is defined to assess the quality of the generated images with the powerful visual-language model CLIP (Radford et al., 2021). Concretely, we compute the cosine similarity between the clip embedding of the generated images and two sentences, $\{$*A high-quality photo, A low-quality photo*$\}$ respectively. We report a $10^3$ scale of the average differences between the two cosine similarities. Finally, we also adopt Blind/Referenceless Image Spatial Quality Evaluator (BRISQUE) (Mittal et al., 2012), a no-reference image quality assessment metric, to evaluate the image quality. For each LDM, we generate 20 images with the 3 prompts respectively $\{$A photo of sks person, A selfie of sks person, A DSLR portrait of sks person$\}$ and report averaged results over the 60 generated images. When generating data from the diffusion models, we use 50-step DDPM sampling and set the negative prompt to be *noisy, low quality, artifacts, poorly drawn face*.

The intuition behind us selecting the $c_{explo}$ resulting in the highest FDS to represent the performance in the prompt mismatch case is that higher FDS means the generated images have higher facial similarity with the training data, which is usually a sign for good fine-tuning result.

## A.2. Data

We mainly adopt the CelebA-HQ (Liu et al., 2015) dataset in our experiments. The dataset consists of thousands of celebrities, with approximately 10 photos each, and the image size is 512x512. We randomly draw 10 celebrities from the dataset and choose 4 images out of every set of images. Normally, $4 \sim 5$ images are sufficient for the LDMs to grasp the core concept in the images (Ruiz et al., 2023; Gal et al., 2023). For experiments on VGGFACE (Cao et al., 2018), we also randomly draw 10 instances from the dataset and choose 4 images out of every set of images.

## A.3. Fine-tuning Hyper-parameter

We fine-tune the Stable Diffusion (Rombach et al., 2022) models with DreamBooth (Ruiz et al., 2023) and LoRA. In Table 8, we list the specific fine-tuning details.

## A.4. Defense Hyper-parameter

We mainly consider three defense algorithms in this paper AdvDM (Liang et al., 2023), FSGM, and ASPL (Van Le et al., 2023). For AdvDM (Liang et al., 2023), we implement the protection loss as $L_{semantic} + \lambda L_{textural}$ with $\lambda = 0.05$, where the $L_{semantic}$ is defined as the training loss of the Stable Diffusion and the textural loss is defined as the $\ell_2$ distance between the mean value of the latent distributions. We run the algorithm for 100 steps (the default number of steps is 40 in the paper) with step size equaling $\varepsilon/10$. For FSGM, we run the defense for 100 iterations, with step size being $\varepsilon/10$ and 1-step gradient accumulation. For ASPL, we run the defense for 50 iterations, for each iteration, the surrogate model is updated for 3 steps and the perturbations are updated for 6 steps. The step size is also set to be $\varepsilon/10$. We set the reference images for training the surrogate models of ASPL and FSGM the same as the images to be protected. Note that we by no means intentionally run the algorithms for fewer steps to boost our proposed defense. The steps are recommended by the original as the default setting and running more steps will not improve their performance significantly. In all, for fair comparison and time consideration, we adopt their default configurations when it comes to the iterations to run.

## A.5. Fine-tuning Prompts when $c_{prot} \neq c_{explo}$

The $c_{prot}$ is fixed to be *A photo of sks person*. We consider 3 $c_{explo}$ when $c_{prot} \neq c_{explo}$ and we encourage future works to try on more diverse prompts or even using soft prompts that are optimized for specific goals for fine-tuning. The exhaustive list of which is

*Table 8.* Fine-tuning hyper-parameters.

| Version | Freeze-TE | LR | Steps | Batch Size | Grad. Accu. | Output Res. |
|---|---|---|---|---|---|---|
| 1.5 | Yes | 2e-6 | 1000 | 1 | 1 | 512x512 |
| 1.5 | No | 2e-6 | 500 | 1 | 1 | 512x512 |
| 2.1 | Yes | 1e-5 | 1000 | 1 | 1 | 728x728 |
| 2.1 | No | 2e-6 | 500 | 1 | 1 | 728x728 |

(a) Fine-tuning hyper-parameters for Dreambooth on CelebA-HQ.

| Version | Freeze-TE | LR | Steps | Batch Size | Grad. Accu. | Output Res. |
|---|---|---|---|---|---|---|
| 1.5 | Yes | 2e-6 | 1200 | 1 | 1 | 512x512 |
| 1.5 | No | 2e-6 | 1000 | 1 | 1 | 512x512 |
| 2.1 | Yes | 1e-5 | 1200 | 1 | 1 | 728x728 |
| 2.1 | No | 2e-6 | 1000 | 1 | 1 | 728x728 |

(b) Fine-tuning hyper-parameters for Dreambooth on VGGFACE.

| Version | Rank | LR | Steps | Batch Size | Grad. Accu. | Output Res. |
|---|---|---|---|---|---|---|
| 1.5 | 16 | 1e-4 | 800 | 2 | 1 | 512x512 |
| 2.1 | 32 | 1e-4 | 1600 | 2 | 1 | 728x728 |

(c) Fine-tuning hyper-parameters for LoRA on CelebA-HQ. We apply the low-rank adapters to both the UNet and the text-encoder.

- *A photo of sks person.*
- *A photo of sks face.*
- *A DSLR portrait of sks person.*

## B. More Results for The Prompt-Mismatch Scnenario

### B.1. Qualitive Results

To better illustrate the performance degradation of the current defense algorithm under the prompt-mismatch scenario, i.e. $\boldsymbol{c_{prot}} \neq \boldsymbol{c_{explo}}$, we randomly select an instance from the CelebA-HQ(Liu et al., 2015) dataset and protect the images with three symbolic defense algorithms, AdvDM (Liang et al., 2023), FSGM (Van Le et al., 2023) and ASPL (Van Le et al., 2023). For each defense, we display 5 images generated by the Stable Diffusion v1.5 (SD v1.5) model fine-tuned on the data protected by it. We display the case where the text-encoder is frozen and unfrozen during fine-tuning in Figure 7a and 7b respectively.

### B.2. Discussion on the Prompt-dependent Effect

The intuition behind the prompt dependency lies in the training loss of the LDM, which requires a textual condition $\boldsymbol{c_{prot}}$ as one of its parameters. Existing attacks all involve the training loss as part of the defense objective to optimize, which makes the resulting perturbations inevitably related to the condition $\boldsymbol{c_{prot}}$.

A very straightforward way to break the correlation between the conditions and the perturbations is to aggregate across several prompts during optimization. However, aggregating through $k$ prompts increases the computational cost by $k$ times and makes the objective even harder to solve. In Table 10, we show that a naive ensembling of prompts to generate the perturbations won't fundamentally resolve the issue.

From Figure 8 to Figure 9, we provide more detailed results to discuss the effect of fine-tuning prompts on single instances.

### B.3. JPEG Results

JPEG compression has long been known to have an undiminishable effect on the perturbations added to images (Aydemir et al., 2018). We show quantitative and qualitative results for the cases where the protected images are saved in JPEG format in Table 10, Figure 10a, and Figure 10b respectively.

*Table 9.* Solving the prompt dependency with aggregation through several prompts during optimization. Here ASPL-k means we aggregate through k prompts. The fine-tuned model is Stable Diffusion v1.5 with text-encoder frozen and the dataset is CelebA-HQ. It can be seen that aggregation through more prompts doesn't bring significant improvement for the prompt-mismatch case. Since the exploiters theoretically have an infinite number of prompts to choose from, we cannot rely on such a method to achieve prompt-independent protection.

| | FDS (↓) | FID (↑) | IQS (↓) | BRISQUE (↑) |
|---|---|---|---|---|
| ASPL-1 | **0.385** | 229.973 | -3.506 | 17.987 |
| ASPL-2 | 0.407 | 250.954 | -3.309 | 18.231 |
| ASPL-4 | 0.401 | **255.728** | **-4.064** | **19.596** |

*Table 10.* The quantitative results showing the performance of the prompt dependent defenses in cases where textual prompts are matched ($c_{prot} = c_{explo}$) and mismatched ($c_{prot} \neq c_{explo}$) between the protection stage and the exploration stage. **Freeze-TE** means whether we Freeze-the-Text-Encoder during fine-tuning. The arrows symbolize the direction of better protection. Perturbed images are saved in JPEG format in this table.

| Freeze-TE | Data | FDS (↓) | FID (↑) | IQS (↓) | BRISQUE (↑) |
|---|---|---|---|---|---|
| ✓ | Clean | 0.480 | 144.570 | 4.130 | 14.757 |
| | AdvDM: $c = c_1$ | 0.386 | 229.973 | -5.768 | 25.340 |
| | AdvDM: $c \neq c_1$ | 0.421 | 157.750 | -4.962 | 20.102 |
| | FSGM: $c = c_1$ | 0.390 | 225.208 | -3.547 | 24.042 |
| | FSGM: $c \neq c_1$ | 0.424 | 211.879 | -1.771 | 16.650 |
| | ASPL: $c = c_1$ | 0.354 | 267.039 | -6.644 | 27.983 |
| | ASPL: $c \neq c_1$ | 0.385 | 229.973 | -3.506 | 17.987 |
| ✗ | Clean | 0.557 | 128.870 | 7.104 | 18.445 |
| | AdvDM: $c = c_1$ | 0.275 | 199.338 | -4.866 | 25.723 |
| | AdvDM: $c \neq c_1$ | 0.377 | 182.473 | -4.286 | 17.857 |
| | FSGM: $c = c_1$ | 0.327 | 258.643 | -8.261 | 22.822 |
| | FSGM: $c \neq c_1$ | 0.412 | 206.979 | -3.316 | 21.238 |
| | ASPL: $c = c_1$ | 0.266 | 311.892 | -9.727 | 22.019 |
| | ASPL: $c \neq c_1$ | 0.340 | 254.840 | -7.588 | 20.326 |

# C. Quantitive Results for Additional Datasets and Training Algorithm

## C.1. LoRA Resutls

We report the results of using LoRA (Hu et al., 2021) as the fine-tuning algorithm on the CelebA-HQ (Liu et al., 2015) dataset in Table 11. We apply the low-rank adapters to both the UNet (Ronneberger et al., 2015) and the CLIP (Radford et al., 2021) text-encoder.

*Table 11.* Evaluation of defense algorithms under the $c_{prot} = c_{explo}$ and $c_{prot} \neq c_{explo}$ scenarios when using LoRA (Hu et al., 2021) for fine-tuning. The best-performing defense under each metric is marked with **bold**.

| Scenario | Data | Stable Diffusion v1.5 | | | | Stable Diffusion v2.1 | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | FDS(↓) | FID(↑) | IQS(↓) | BRISQUE(↑) | FDS(↓) | FID(↑) | IQS(↓) | BRISQUE(↑) |
| | Clean | 0.465 | 224.829 | 7.353 | 13.116 | 0.459 | 222.446 | 14.581 | 7.977 |
| $c_{prot} = c_{explo}$ | AdvDM | 0.330 | 300.942 | -10.414 | 17.591 | 0.219 | 354.107 | -3.662 | 35.402 |
| | FSGM | 0.309 | **373.485** | -9.524 | 11.053 | 0.173 | 423.777 | -12.004 | 41.896 |
| | ASPL | 0.295 | 372.883 | -8.113 | 15.165 | 0.174 | 376.804 | -4.950 | 40.661 |
| | PID | **0.231** | 341.410 | **-18.782** | **46.977** | **0.163** | **414.323** | **-17.694** | **59.272** |
| $c_{prot} \neq c_{explo}$ | AdvDM | 0.377 | 283.286 | -3.178 | 14.504 | 0.239 | 338.788 | -4.998 | 33.312 |
| | FSGM | 0.334 | 362.437 | -3.800 | 10.633 | 0.240 | 391.125 | -8.771 | 34.302 |
| | ASPL | 0.327 | **381.953** | -4.210 | 13.613 | 0.225 | 352.907 | -4.033 | 38.890 |
| | PID | **0.276** | 322.451 | **-11.353** | **36.335** | **0.212** | **396.638** | **-18.327** | **50.168** |

## C.2. VGGFACE Resutls

We report the results for the VGGFACE dataset (Cao et al., 2018) in Table 12.

*Table 12.* Quantitive evaluation of defense algorithms under the $c_{prot} = c_{explo}$ and $c_{prot} \neq c_{explo}$ scenarios on the VGGFACE dataset. The best-performing defense under each metric is marked with **bold**.

| Scenario | Data | Frozen Text Encoder | | | | Unfrozen Text Encoder | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | FDS($\downarrow$) | FID($\uparrow$) | IQS($\downarrow$) | BRISQUE($\uparrow$) | FDS($\downarrow$) | FID($\uparrow$) | IQS($\downarrow$) | BRISQUE($\uparrow$) |
| | Clean | 0.442 | 164.203 | 16.359 | 12.290 | 0.512 | 134.21 | 17.63 | 7.37 |
| $c_{prot} = c_{explo}$ | AdvDM | 0.332 | 211.935 | 0.928 | 13.853 | 0.244 | 235.402 | -5.694 | 21.423 |
| | FSGM | 0.294 | **282.948** | 0.064 | 19.111 | 0.235 | 285.889 | **-10.024** | 23.694 |
| | ASPL | 0.312 | 274.449 | **-3.710** | 22.479 | 0.266 | 293.886 | -6.178 | 26.454 |
| | PID | **0.203** | 278.809 | -1.301 | **26.772** | **0.223** | **295.264** | -4.730 | **27.273** |
| $c_{prot} \neq c_{explo}$ | AdvDM | 0.348 | 222.290 | -0.317 | 12.416 | 0.329 | 214.461 | 1.853 | 16.445 |
| | FSGM | 0.321 | 259.285 | -2.736 | 17.554 | 0.329 | 232.522 | **-7.382** | 18.067 |
| | ASPL | 0.326 | **279.141** | **-2.767** | **21.150** | 0.350 | 235.014 | -1.329 | 14.721 |
| | PID | **0.230** | 273.189 | -1.577 | 20.622 | **0.247** | **300.853** | -3.595 | **24.352** |

(a) The fine-tuned model is Stable Diffusion v1.5.

| Scenario | Data | Frozen Text Encoder | | | | Unfrozen Text Encoder | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | FDS($\downarrow$) | FID($\uparrow$) | IQS($\downarrow$) | BRISQUE($\uparrow$) | FDS($\downarrow$) | FID($\uparrow$) | IQS($\downarrow$) | BRISQUE($\uparrow$) |
| | Clean | 0.455 | 227.395 | 15.580 | 10.413 | 0.482 | 200.339 | 13.935 | 7.524 |
| $c_{prot} = c_{explo}$ | AdvDM | 0.241 | **301.904** | 2.202 | 22.885 | 0.321 | 252.942 | **-7.852** | 26.315 |
| | FSGM | **0.236** | 275.736 | **-5.329** | **34.469** | **0.253** | **342.073** | -7.091 | 28.114 |
| | ASPL | 0.253 | 280.514 | 0.827 | 27.108 | 0.302 | 251.864 | -6.905 | **29.900** |
| | PID | 0.301 | 282.013 | 1.749 | 15.406 | 0.321 | 267.520 | -1.258 | 27.993 |
| $c_{prot} \neq c_{explo}$ | AdvDM | **0.269** | **284.283** | -0.553 | 17.726 | 0.364 | 219.191 | -1.257 | 24.797 |
| | FSGM | 0.297 | 269.976 | **-9.208** | **32.260** | 0.351 | 226.134 | 1.560 | **29.667** |
| | ASPL | 0.296 | 240.470 | 1.612 | 25.668 | 0.340 | 234.291 | **-5.408** | 27.554 |
| | PID | 0.338 | 267.284 | 3.039 | 14.360 | **0.337** | **252.355** | 0.066 | 25.186 |

(b) The fine-tuned model is Stable Diffusion v2.1.

### C.3. $8/255$ Resutls

We report the results of applying a tighter constraint on the perturbation budget, i.e. $\varepsilon_\infty = 8/255$ when running the defensive algorithms on the CelebA-HQ (Liu et al., 2015) dataset in Table 13. The fine-tuned model is Stable Diffusion v1.5.

## D. Influence on Image Quality

In this section, we assess the influence of PID on the image quality. We adopt three metrics to quantitively measure the influence. The three metrics are SSIM (Wang et al., 2004), PSNR, and LPIPS (Zhang et al., 2018) [6]. We report the quantitive results in Table 14 and the qualitative comparisons are provided in Figure 11. PID affects the images in a more aggressive way, which might contribute to its better robustness against image corruption and transferability across models.

## E. Disscusion on PID and Image Editing

Zero-shot image editing (Meng et al., 2021; Kawar et al., 2023) is another amazing ability of the LDMs and can also threaten civil privacy. Following the setting of Salman et al. (2023), we test the performance of PID in the image editing scenario. We protect the default image provided by Salman et al. (2023) with the simple attack proposed by it, which targetedly manipulates the mean value and our defense. The model is the Stable Diffusion Inpainting [7] and both the attacks are obtained via PGD$_{100}$. The perturbation budget is $\varepsilon_\infty = 0.05$.

Observing Figure 12, the PID has a larger influence on the inpainting result than the simple attack, injecting meaningless noise and disrupting the image's semantics. However, we find that PID is yet not capable enough to adequately protect the images in this scenario, with the image still largely plausible. We leave the application of prompt-independent defense in the

[6] We adopt the implementation from https://github.com/photosynthesis-team/piq

[7] Downloaded from https://huggingface.co/runwayml/stable-diffusion-inpainting

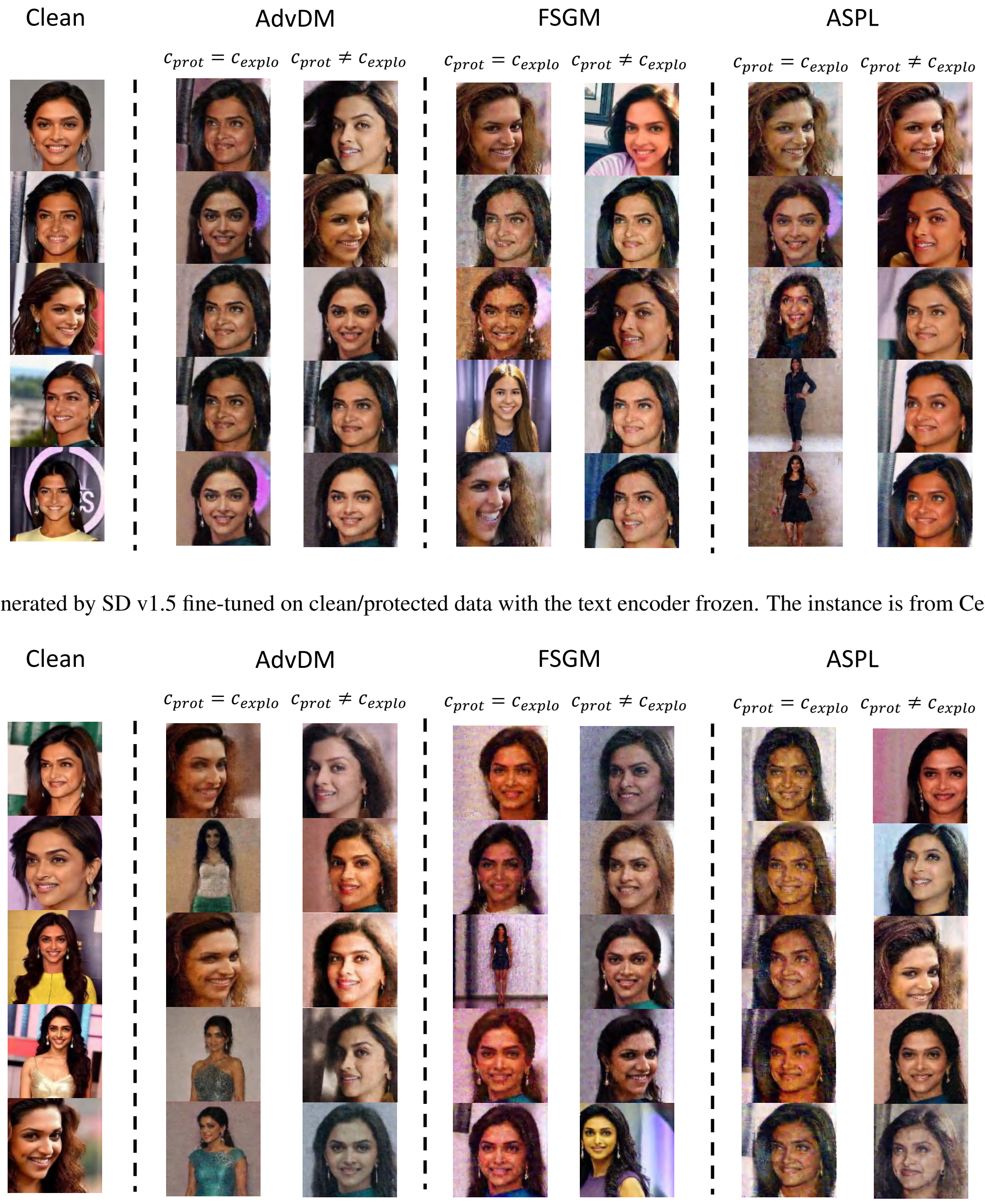
*Table 13.* Evaluation of defense algorithms under the $\boldsymbol{c_{prot}} = \boldsymbol{c_{explo}}$ and $\boldsymbol{c_{prot}} \neq \boldsymbol{c_{explo}}$ scenarios under the constraint that $\varepsilon_\infty = 8/255$. The fine-tuned model is SD v1.5. The best-performing defense under each metric is marked with **bold**.

| Freeze-TE | Data | FDS (↓) | FID (↑) | IQS (↓) | BRISQUE (↑) |
|---|---|---|---|---|---|
| | Clean | 0.480 | 144.570 | 4.130 | 14.757 |
| ✓ | AdvDM | 0.375 | 209.807 | -5.405 | 21.344 |
| | FSGM | 0.403 | 214.184 | -2.998 | 24.093 |
| | ASPL | 0.412 | 209.987 | -4.213 | 23.020 |
| | PID | **0.256** | **248.038** | **-9.813** | **33.342** |
| ✗ | AdvDM | 0.410 | **227.546** | -5.572 | 14.913 |
| | FSGM | 0.429 | 200.422 | -1.174 | 22.564 |
| | ASPL | 0.416 | 200.152 | -3.266 | 21.091 |
| | PID | **0.312** | 221.636 | **-9.949** | **27.391** |

*Table 14.* The influence of defensive perturbations on image quality with three matrices measuring the difference between the perturbed images and the clean images.

| | SSIM | PSNR | LPIPS |
|---|---|---|---|
| Random Noise | 0.82 | 30.90 | 0.20 |
| AdvDM | 0.85 | 32.57 | 0.19 |
| FSGM | 0.91 | 35.76 | 0.19 |
| ASPL | 0.90 | 35.91 | 0.19 |
| PID | 0.71 | 30.90 | 0.20 |

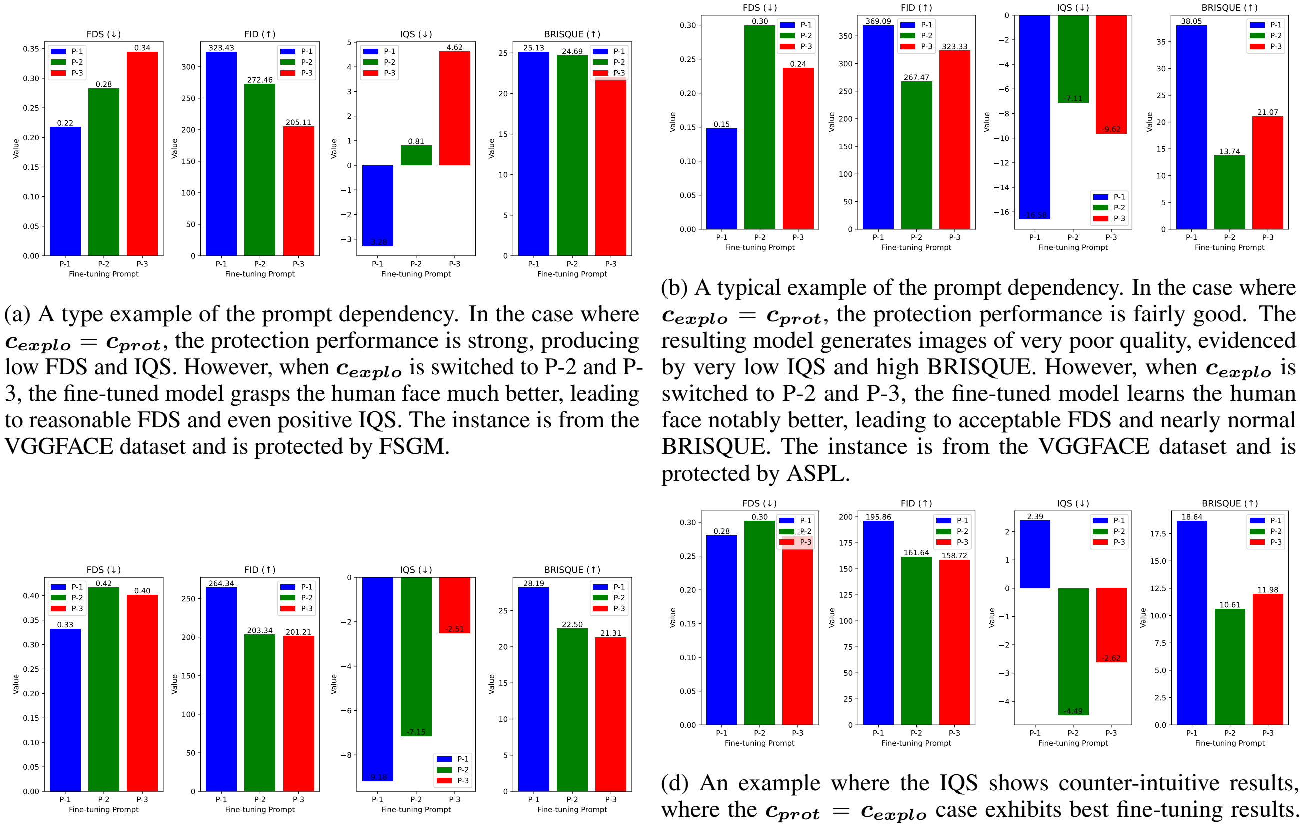
image inpainting scenario for future work.



(a) Images generated by SD v1.5 fine-tuned on clean/protected data with the text encoder frozen. The instance is from CelebA-HQ.

(b) Images generated by SD v1.5 fine-tuned on clean/protected data with the text encoder unfrozen. The instance is from CelebA-HQ.

*Figure 7.* Influence of prompt-mismatch, i.e., $\boldsymbol{c_{prot}} \neq \boldsymbol{c_{explo}}$, on the protective performance.



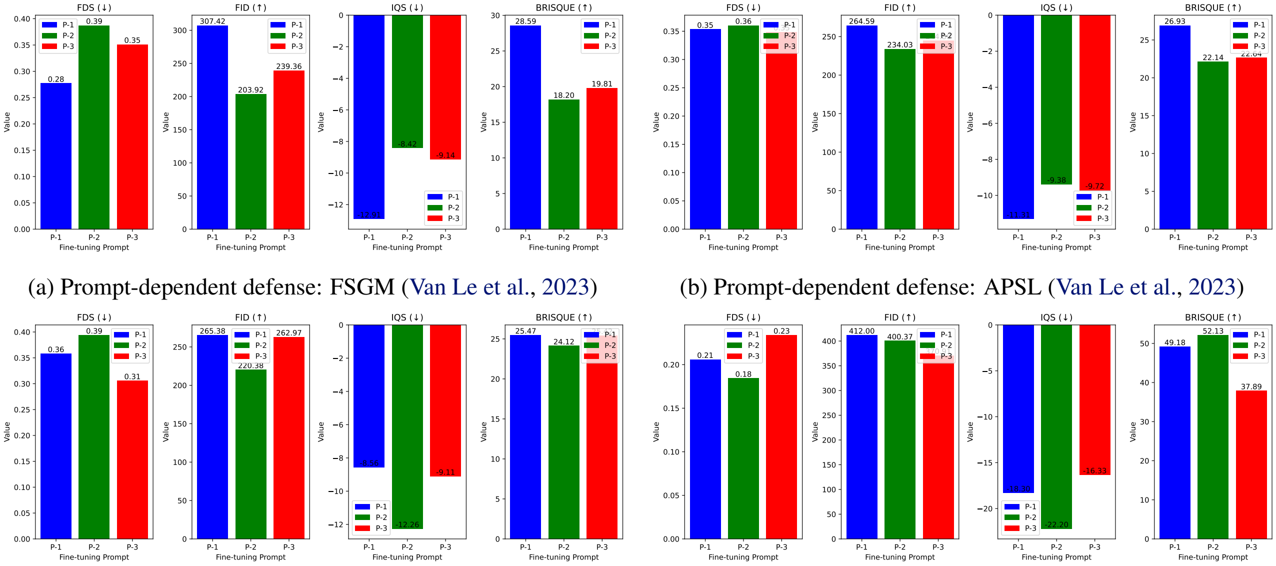
(a) A type example of the prompt dependency. In the case where $c_{explo} = c_{prot}$, the protection performance is strong, producing low FDS and IQS. However, when $c_{explo}$ is switched to P-2 and P-3, the fine-tuned model grasps the human face much better, leading to reasonable FDS and even positive IQS. The instance is from the VGGFACE dataset and is protected by FSGM.

(b) A typical example of the prompt dependency. In the case where $c_{explo} = c_{prot}$, the protection performance is fairly good. The resulting model generates images of very poor quality, evidenced by very low IQS and high BRISQUE. However, when $c_{explo}$ is switched to P-2 and P-3, the fine-tuned model learns the human face notably better, leading to acceptable FDS and nearly normal BRISQUE. The instance is from the VGGFACE dataset and is protected by ASPL.

(c) An example where the perturbations exhibit medium-level dependency on prompts. The protection is slightly weaker for P-2 and P-3. The instance is from the VGGFACE dataset and is protected by ASPL.

(d) An example where the IQS shows counter-intuitive results, where the $c_{prot} = c_{explo}$ case exhibits best fine-tuning results. Even though, the dependency effect is still valid across the other three metrics. We by no means claim that $c_{prot} \neq c_{explo}$ always leads to a significant performance drop on every instance protected by any prompt-dependent defense. As shown in our quantitative results, our claim should be interpreted as an overall trend rather than a definite rule. The instance is from the VGGFACE dataset and is protected by FSGM.

*Figure 8.* instances selected to illustrate the effect of the mismatch between $c_{prot}$ and $c_{explo}$. P-1, P-2, and P-3 denote three $c_{explo}$ defined in A and $c_{prot}$ is still *A photo of sks person*. Note that these sub-figures serve as case studies rather than rigorous evidence for our claims. The fine-tuned model is SD v1.5 with text-encoder unfrozen.



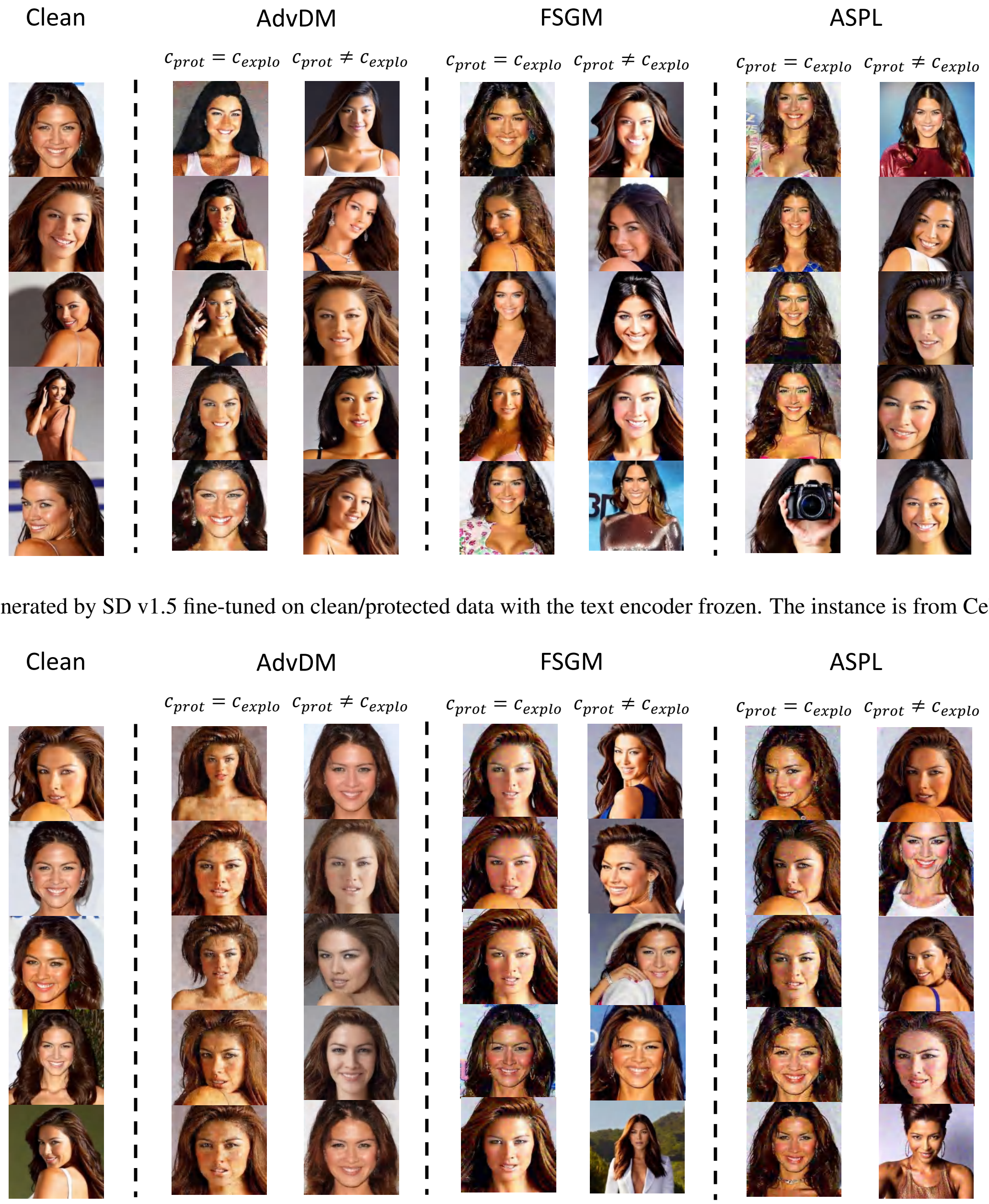
(a) Prompt-dependent defense: FSGM (Van Le et al., 2023)

(b) Prompt-dependent defense: APSL (Van Le et al., 2023)

(c) Prompt-dependent defense: AdvDM (Liang et al., 2023)

(d) Prompt-dependent defense: PID

*Figure 9.* Averaged fine-tuning results across the CelebA-HQ dataset for different $\boldsymbol{c_{explo}}$. For ASPL and FSGM, the defense performance in the $\boldsymbol{c_{prot}} = \boldsymbol{c_{explpo}}$ case out-performs the $\boldsymbol{c_{prot}} \neq \boldsymbol{c_{explo}}$ case, as evidenced by all four evaluation metrics. For AdvDM, which has a prompt-independent component, $L_{textural}$, as part of its defense target, better resilience against varied prompts is observed, as evident by P-3 achieving worse fine-tuning results than P-1. Contrary to the prompt-dependent defenses, the performance of PID remains steady when different fine-tuning prompts are used, which benefits from the perturbations' independence of the textual condition. The fine-tuned model is SD v1.5 with text-encoder unfrozen.



(a) Images generated by SD v1.5 fine-tuned on clean/protected data with the text encoder frozen. The instance is from CelebA-HQ.

(b) Images generated by SD v1.5 fine-tuned on clean/protected data with the text encoder unfrozen. The instance is from CelebA-HQ.

*Figure 10.* Influence of prompt-mismatch, i.e., $c_{prot} \neq c_{explo}$, on the protective performance. Protected images are saved in JPEG format.

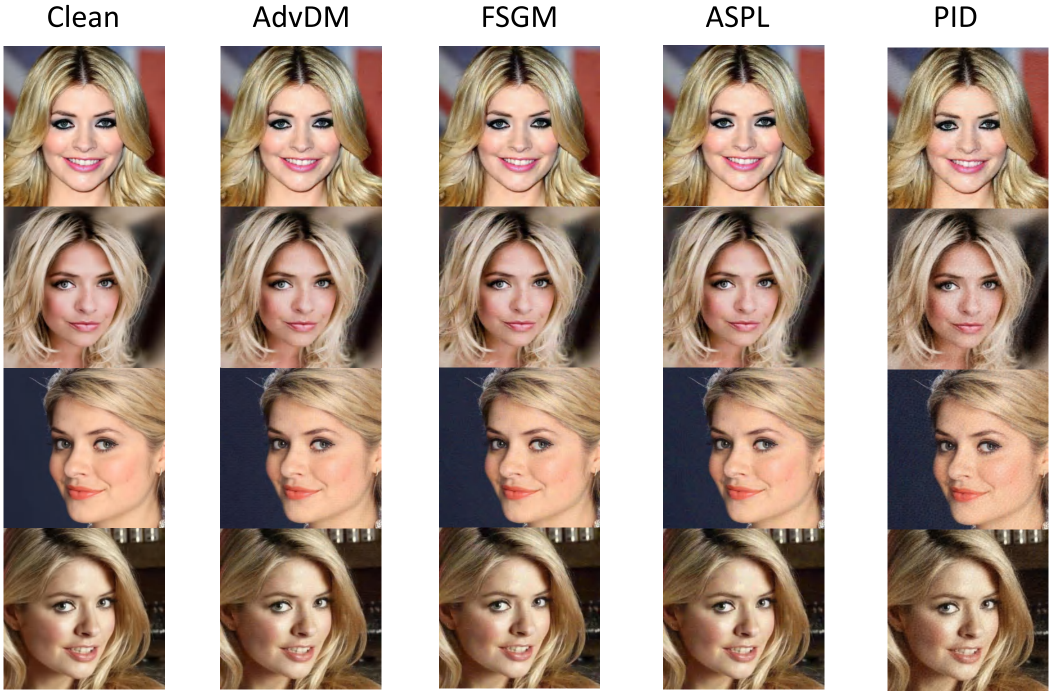
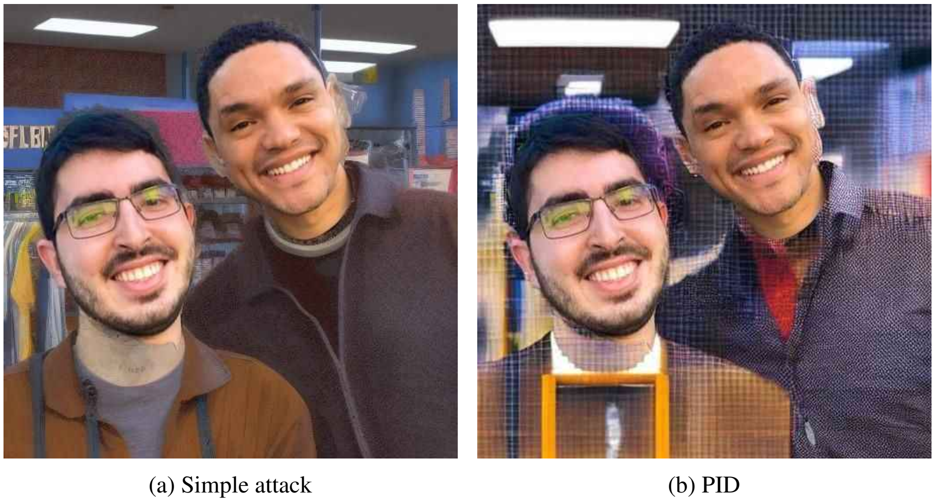


*Figure 11.* Comparision between the clean images and images protected by the four defensive algorithms. We use the SD v1.5 to generate these images. The pertubation budget is $\varepsilon_\infty = 8/255$.



(a) Simple attack

(b) PID

*Figure 12.* PID's performance in the image editing scenario following the setting of Salman et al. (2023). (a) the simple attack in Salman et al. (2023). (b) PID. The inference prompt is *two men in a library*.

## F. More Visualization

**CAUTION: The images presented below may cause DISCOMFORT.**

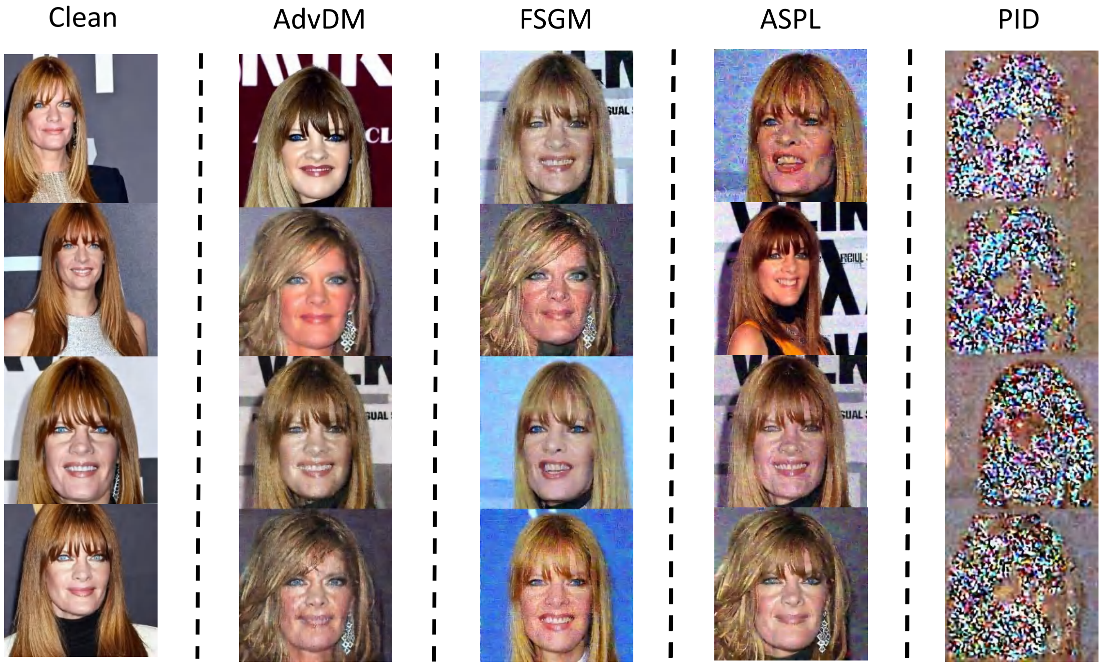


*Figure 13.* Images synthesized by SD v1.5 fine-tuned on an instance from the CelebA-HQ protected by the defensive algorithms. The text-encoder is frozen during fine-tuning and the algorithm is Dreambooth.

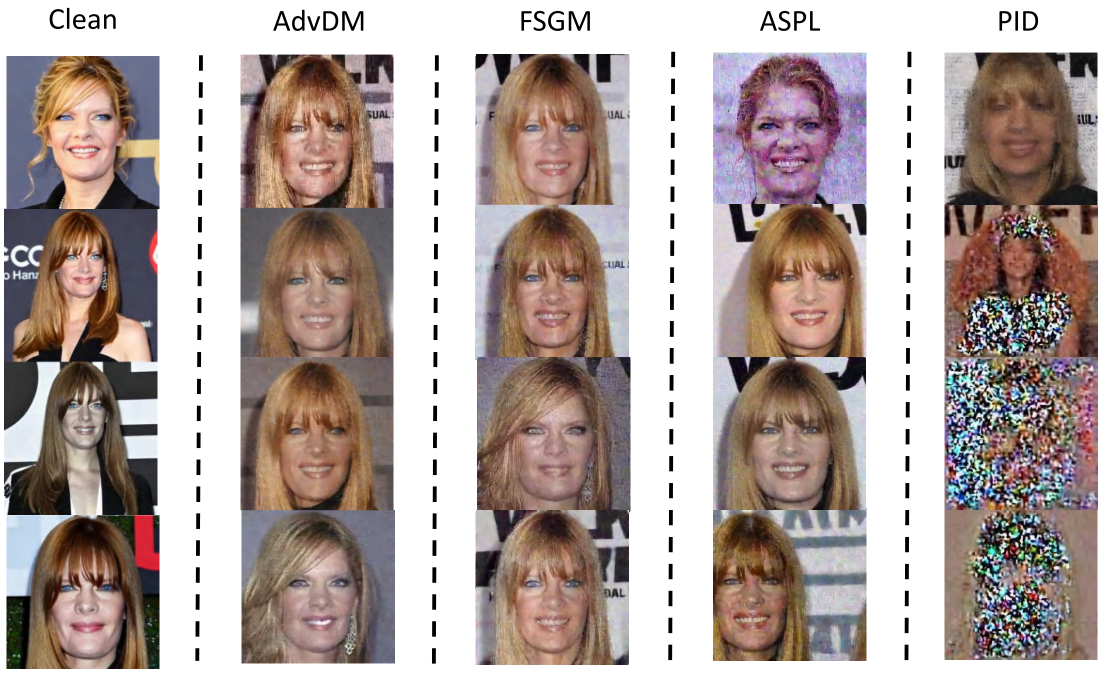


*Figure 14.* Images synthesized by SD v1.5 fine-tuned on an instance from the CelebA-HQ protected by the defensive algorithms. The text-encoder is trained during fine-tuning and the algorithm is Dreambooth.

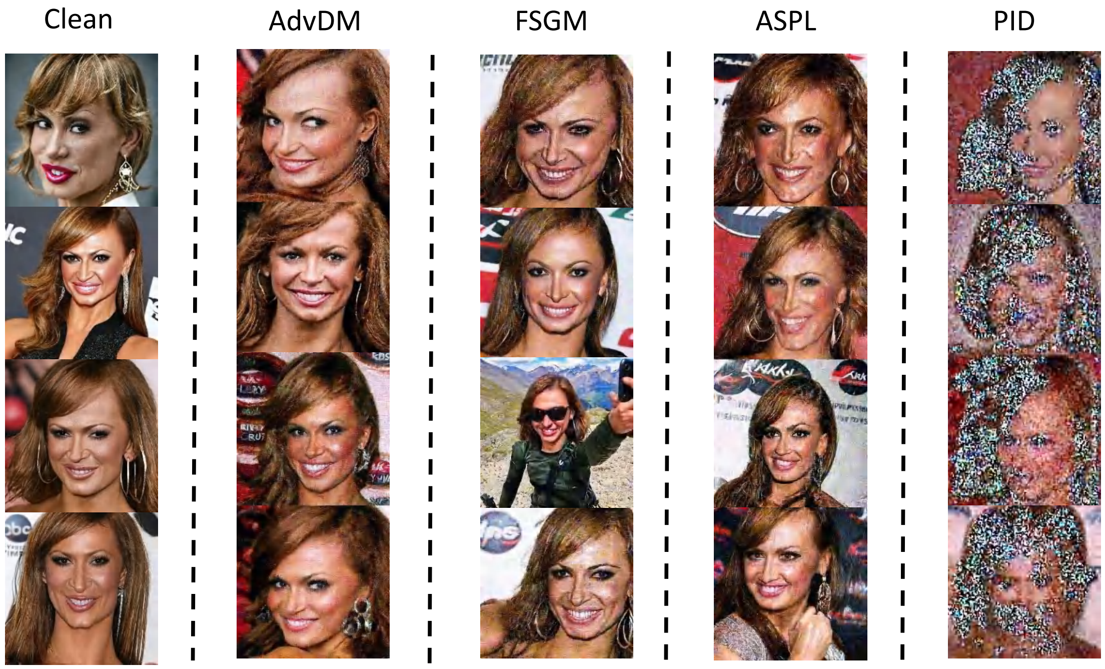


*Figure 15.* Images synthesized by SD v2.1 fine-tuned on an instance from the CelebA-HQ protected by the defensive algorithms. The text-encoder is frozen during fine-tuning and the algorithm is Dreambooth.

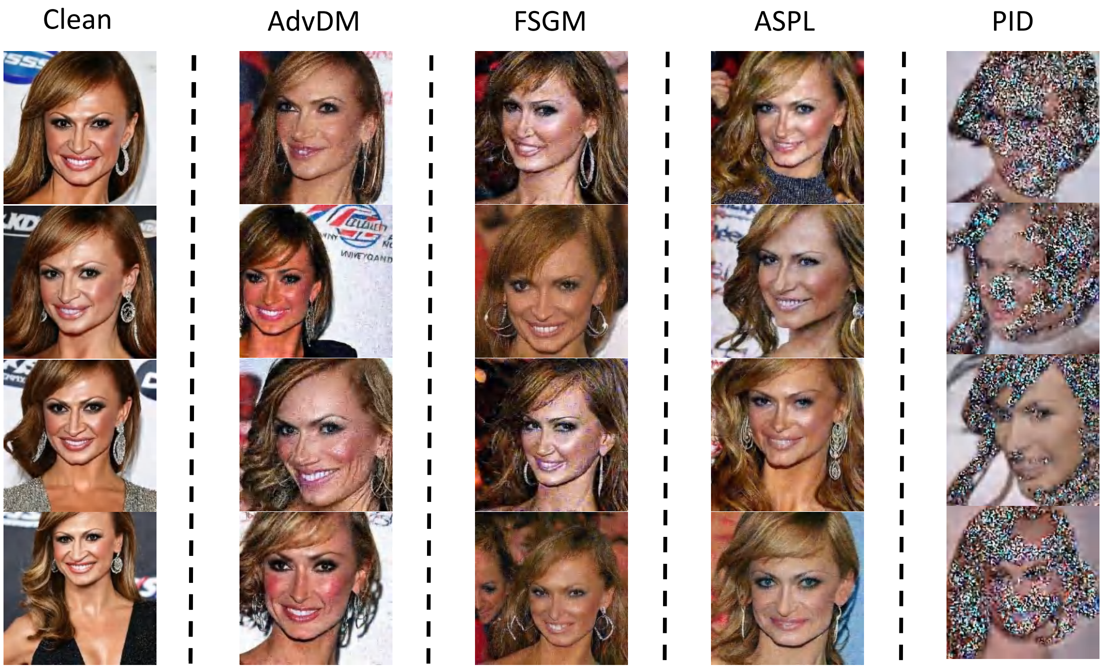


*Figure 16.* Images synthesized by SD v2.1 fine-tuned on an instance from the CelebA-HQ protected by the defensive algorithms. The text-encoder is trained during fine-tuning and the algorithm is Dreambooth.

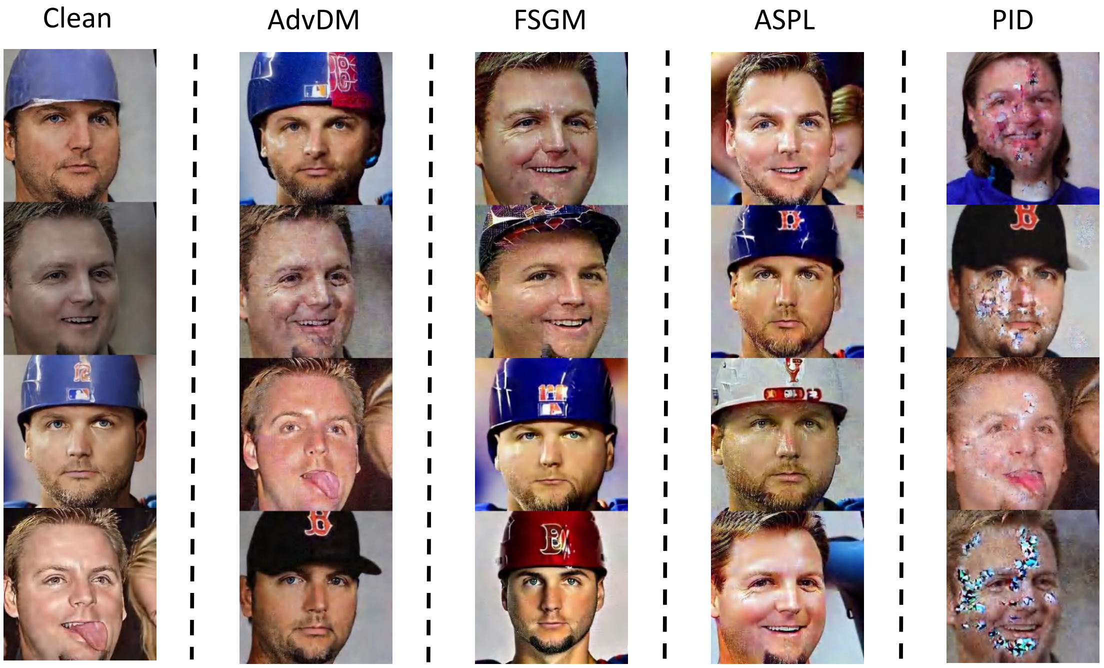


*Figure 17.* Images synthesized by SD v1.5 fine-tuned on an instance from the VGGFACE protected by the defensive algorithms. The text-encoder is frozen during fine-tuning and the algorithm is Dreambooth.

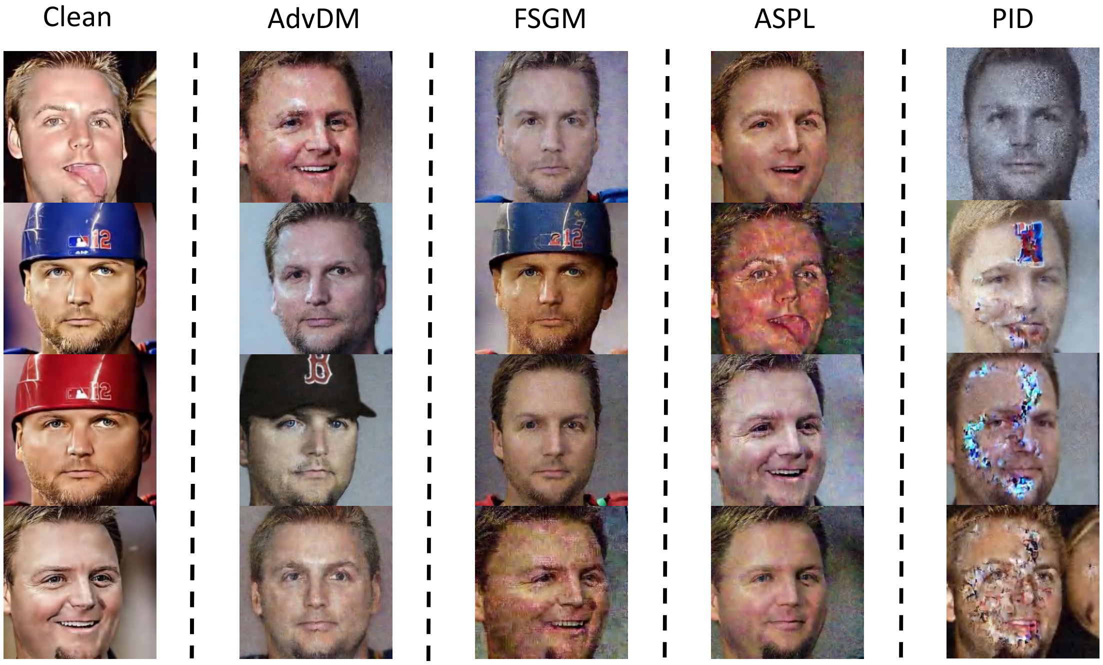


*Figure 18.* Images synthesized by SD v1.5 fine-tuned on an instance from the VGGFACE protected by the defensive algorithms. The text-encoder is trained during fine-tuning and the algorithm is Dreambooth.

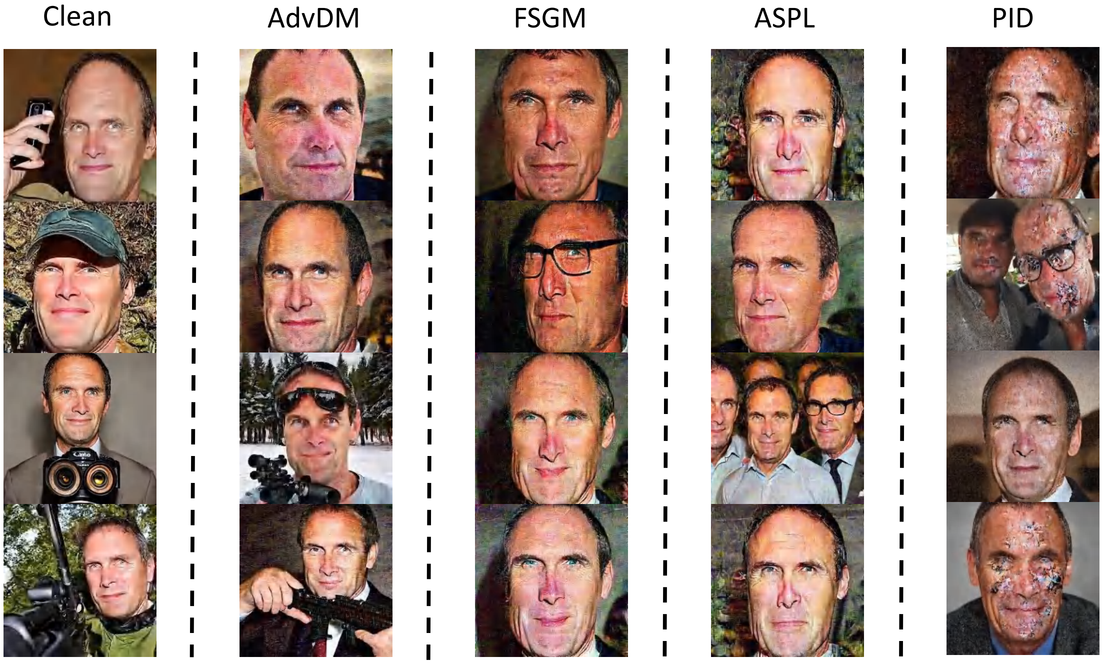


*Figure 19.* Images synthesized by SD v2.1 fine-tuned on an instance from the VGGFACE protected by the defensive algorithms. The text-encoder is frozen during fine-tuning and the algorithm is Dreambooth.

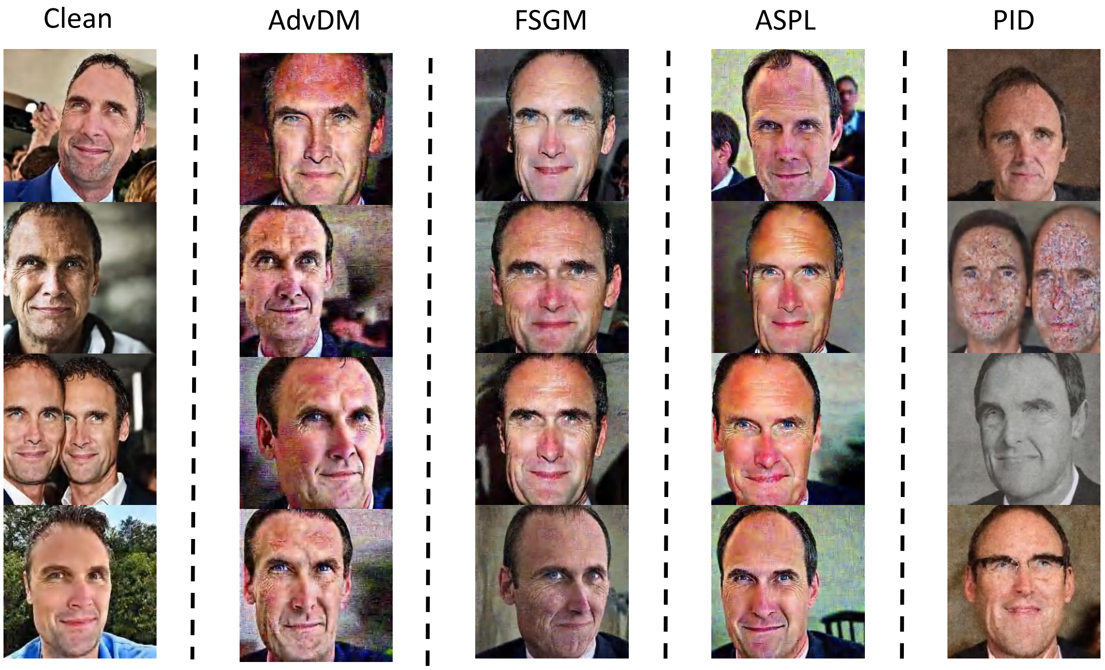


*Figure 20.* Images synthesized by SD v2.1 fine-tuned on an instance from the VGGFACE protected by the defensive algorithms. The text-encoder is trained during fine-tuning and the algorithm is Dreambooth.

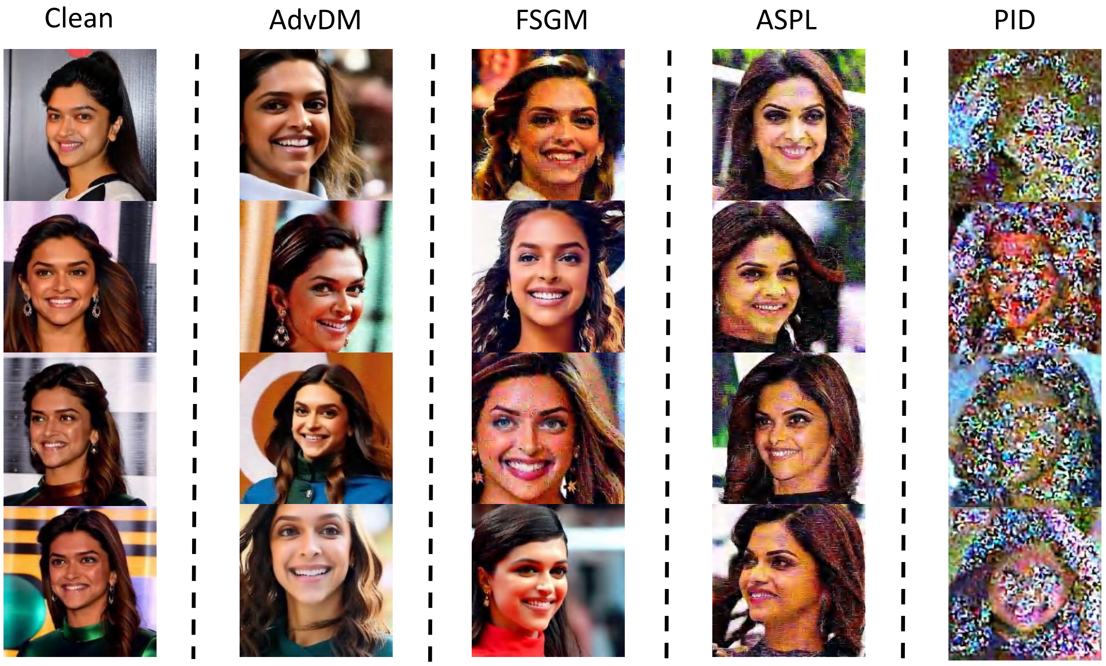


*Figure 21.* Images synthesized by SD v1.5 fine-tuned on an instance from the CelebA-HQ protected by the defensive algorithms. The fine-tuning algorithm is LoRA.

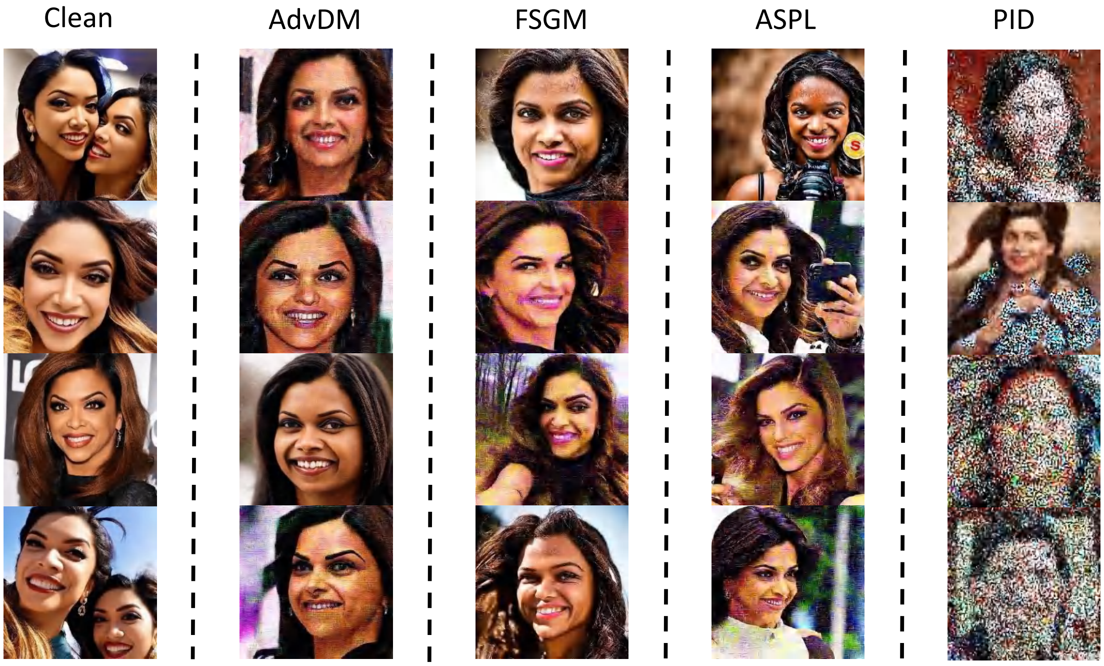


*Figure 22.* Images synthesized by SD v2.1 fine-tuned on an instance from the CelebA-HQ protected by the defensive algorithms. The fine-tuning algorithm is LoRA.