Pretender: Universal Active Defense against Diffusion Finetuning Attacks

Zekun Sun[1], Zijian Liu[1], Shouling Ji[2], Chenhao Lin[3], Na Ruan[1†]
[1]*Shanghai Jiao Tong University,* [2]*Zhejiang University,* [3]*Xi'an Jiaotong University*

Abstract

The proliferation of Diffusion Models (DMs) has marked a significant advancement in AI-generated image creation. However, this success has also spawned a new form of infringement threat termed the Diffusion Finetuning Attack (DFA), where malicious attackers can finetune pre-trained DMs using minimal resources to illicitly synthesize copyright-infringing images by 'stealing' information from personal photographic data or artwork, raising critical concerns about privacy and intellectual property rights. Recognizing the limitations of current defense strategies, which exhibit inadequate generalizability and suboptimal mechanism efficacy, we introduce an universal and effective active defense mechanism that applies subtle protective noise to images, guarding against information theft from DFAs. Our work innovatively conceptualizes active defense as a bi-level optimization problem, focusing on attackers' common behaviors to enhance the generalization of defense. Guided by this optimization framework, we have developed a novel algorithm named `Pretender`, where we adversarially trained a surrogate model to facilitate the generation of more effective protective noise. In addition, a Simultaneous Gradient Back-Propagation (SGBP) technique is introduced to significantly enhance computational efficiency. Extensive experiments including real-world evaluations have demonstrated the effectiveness of `Pretender`. By applying minimal perturbations (p = 0.03), `Pretender` successfully disrupted the quality and semantics of images synthesized by diverse DFAs, achieving a comprehensive and prominent improvement in various automated evaluation metrics by 22.27% and in human assessment scores by 94.28%. [1]

1 Introduction

The Diffusion Model **(DM)** [1] has recently ignited a surge in AI-generated image creation. The model displays exceptional performance in terms of high resolution, photo-realistic quality, and remarkable diversity, achieving a level comparable to that of human professionals in the fields of photography, painting, and design [2]. Its technology is constantly evolving, with projects such as Midjourney [3] and Stable Diffusion [4] leading the way.

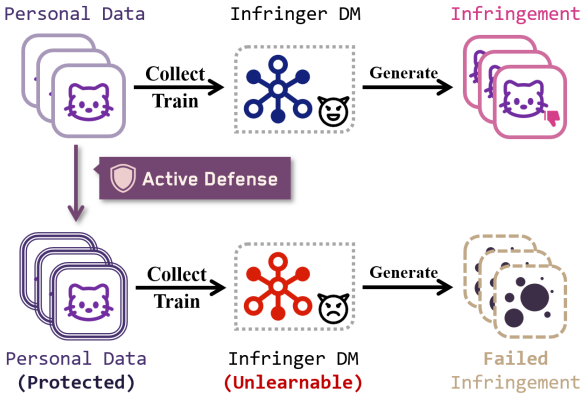

Figure 1: Active Defense against the Diffusion Finetuning Attack (DFA).

However, the superior performance of DMs has also precipitated a highly threatening form of infringement attack: the *Diffusion Finetuning Attack* **(DFA)**. Malicious attackers can exploit efficient algorithms to finetune a pre-trained DM, enabling them to illicitly extract information from personal photographic data or artwork extensively shared on social networks. Based on stolen information, they can synthesize images that may have negative effects without authorization, e.g., fabricate a designated person to create pornographic imagery [5], or mimic a particular visual style of an artist for undue commercial gain [6]. **As of September 2024, more than five finetuning algorithms [7–13] have continuously emerged**, with three typical ones, LoRA (LR) [7], Textual Inversion (TI) [8] and DreamBooth (DB) [9], being extensively used in the open-source community.

The primary threat of DFA lies in its capacity for effective attacks with minimal effort. Using just about five images, the DFA can be launched by tools such as `Diffusers` [14] and `SD-webUI` [15] which enable complex finetuning and im-

†Corresponding author: Na Ruan.
[1]Codes: https://github.com/frederickszk/Pretender

age generation on standard personal computers with minimal code, lowers barriers significantly, thus amplifying the potential for misuse. In response, mainstream research has adopted *active defense* mechanisms to shield personal images from the threat of DFA, such as `Glaze` [16] and `AdvDM` [17]. These methods involve adding imperceptible protective noise to images before they are shared, rendering it difficult for a DFA that is fine-tuned on these images to effectively mimic them.

Research Gap. Despite recent efforts making significant advances in defending against DFA, we have identified several unignorable research gaps. ❶ **Limited generalizability.** Existing active defense strategies struggle to generalize across different DFA algorithms and various attack scenarios. `AdvDM` effectively disrupts TI finetuning attack [8] by identifying adversarial samples for DM but may remain vulnerable to other DFAs. `Glaze` disturbs the style representation of images in the DM's feature space to mislead DFA into stealing incorrect styles. However, it only defends against style theft and cannot protect against the misuse of specific objects, characters, or facial identities, which could be exploited to generate infringing content such as pornography, violence, or misleading images. ❷ **Suboptimal defense mechanisms.** Strategies for effectively disrupting DFA are challenging and underexplored. It requires altering a high-dimensional probability distribution, much more difficult than merely manipulating data-label mapping in common image protection tasks for classification models [18–20] or human-face analysis systems [21, 22]. Aforementioned DFA defense methods [16, 17] both utilize a pre-trained DM as a *surrogate model* to compute protective noise. Nevertheless, this *static* surrogate model fails to faithfully simulate the complex behavior of attackers, thereby diminishing the defense effectiveness.

Motivation. To bridge these research gaps, we aim to develop a universal and effective active defense algorithm capable of countering various DFAs. Our fundamental insight is that: ❶ **Targeting attackers' collective behaviors yields an universal defense algorithm.** Recent research practices indicate that over-reliance on specific DFA principles may impedes the generalization. To this end, we extract the ubiquitous behavioral pattern of attackers, and innovatively conceptualize active defense as a universal *bi-level optimization* problem theoretically, capable of encompassing multiple DFAs. This theoretical framework reveals the principles of attack-defense interaction and the essential optimization difficulties, shedding light on the design of defense algorithms.

We further employ the broadly adopted and efficacious surrogate model strategy to explore solutions to the active defense optimization problem. Inspired by adversarial training, our motivation lies in that ❷ **An adversarially-robust surrogate model spurs effective protective noise.** By identifying a common limitation in existing approaches that rely solely on a static surrogate model, we propose *dynamically* updating the parameters of the surrogate model to enhance its robustness to protective noise. The final protective noise optimized for this updated surrogate model is anticipated to possess enhanced protective capabilities and transferability.

Following the motivations for coutering the research challenges, we develop a universal and efficient DFA defense algorithm named `Pretender`, which generates stronger protection by adversarially perturbing the samples and synchronously updating the model parameters, enabling it to defend against diverse DFAs. In addition, a novel training strategy named Simultaneous Gradient Back-Propagation (SGBP) is proposed, which optimizes the complex iterative training process and significantly conserves computational resources.

Contribution. Our contributions can be concluded as follows:

- We explore the problem of active defense against a highly threatening infringement attack in the AI-creation era: the Diffusion Finetuning Attack (DFA). Through meticulous analysis of existing challenges, we propose a novel defense algorithm named `Pretender` with ***high generalizability***. It not only safeguards artistic images but also provides protection for characters and identity images, while effectively mitigating diverse DFAs. To the best of our knowledge, ***we are the first to pioneer a universal defense against multiple DFAs***, filling a critical gap in the research field.

- To enhance the generalizability of the defense, we ***summarize the attackers' collective behaviors*** to propose a bi-level optimization objective for defense that has not been addressed by prior works. Building on this, we introduced the ***adversarial-for-unlearnable*** concept, leveraging an adversarially robust surrogate model to improve the transferability of protection. Lastly, we develop a ***novel optimization strategy***, SGBP, which transforms the traditional alternating optimization approach into a continuous optimization process, substantially enhancing time efficiency.

- Extensive and comprehensive experiments demonstrate the effectiveness of our proposed approach. `Pretender` effectively hinders infringers using DFAs from imitating the visual patterns of the protected images, such as specific characters or artistic styles. Furthermore, its robustness, transferability, and real-world applicability have been thoroughly evaluated and validated.

2 Related Work

2.1 Diffusion Finetuning Attack

Diffusion Model: It firstly defines a forward noising process by adding Gaussian noise to image sample x_0, and the generative model is defined as the backward de-noising process, which learns to revert the original image from the noised image [1] . The principle of training is to match each step in adding noise and removing noise. Through the derivation of the Variational Lower Bound and a series of simplifications, the loss function during the training of DM can be written as:

$$L(\theta, x) = \mathbb{E}_{t,x_0,\varepsilon}\left[\|(\varepsilon - \varepsilon_\theta(x_t, t))\|^2\right], \quad (1)$$

where the ε represents the noise added in the forward step, and the ε_θ is the model's prediction in the backward step.

DM excels in both "*Text-to-Image*" (T2I) and "*Image-to-Image*" (I2I) tasks. For T2I, numerous open-source or commercial models like Stable Diffusion [23], DALL·E [24], and Midjourney [3] leverage the capabilities of multi-modal models such as CLIP [25] to generate images from text. In the I2I domain, DM supports tasks such as super-resolution [26], inpainting [27], style transfer [28], and depth map rendering [29], underscoring their extensive utility in image generation tasks.

Finetuning Attacks: The Diffusion Finetuning Attack (DFA) empowers infringers with the capability to imitate the visual patterns of an original image using a small set of collected images, including unique artistic styles or characters. Considering representativeness, open-source adaptability and popularity, our work mainly focuses on the following three finetuning techniques for launching DFAs:

① ***LoRA (LR)*** [7]: It was originally proposed for fast finetuning of Large Language Models (LLM) [30] and has been adapted for DM. LR works by adding a very small number of trainable parameters to the backbone UNet [31] of DM, allowing for quick finetuning of the model parameters.

② ***Textual Inversion (TI)*** [8]: It primarily controls the word embedding stage in the workflow of the DM. The text prompt used to control the image generation is first converted into a word embedding. Thus, the core idea behind TI is to add a trainable embedding, which is bound to a specific visual concept to be learned. This approach enables the utilization of a specific token (embedding) to conduct targeted generation.

③ ***DreamBooth (DB)*** [9]: The core concept behind the DB is the Prior-Preservation-Loss, which draws on the powerful prior of the pre-trained DM to synthesize data of specific classes, usually the same to the collected personal training samples. This auxiliary data is then trained together with the personal training data to prevent overfitting and concept collapse, thus optimizing the attack effect.

In addition, ④ **SVDiff** [12] performs singular value decomposition on the weight matrices of the backbone UNet and finetunes only a limited portion. Similar to ***LR***. ⑤ **Hifi-Tuner** [11] builds on ***TI*** and involves training an extra word embedding. It optimizes synthesis quality by applying mask guidance and regularization. ⑥ **TRL** [13] and ⑦ **DraFT** [10] finetune the model using different reward functions as loss functions, conceptually resembling **DB**, while their implementation is based on **LR** to train the networks. Given that the core training mechanisms of these DFAs are consistent with the typical ones (LR, TI, DB), focusing on the three typical DFAs allows our defense framework to capture and mitigate common threats, suggesting its potential to generalize to other principle-similar DFAs. We provide further discussion in Sec. 11.

2.2 Active Defense for Data Protection

Against Classification Models: Researches on protecting personal images from deep-learning systems are predominantly centered on classification models. This technique is also commonly referred to as availability poisoning attacks or ***unlearnable*** examples [18, 19]. It involves adding trivial protective noise to the images in the training set T that are to be protected, with the aim of degrading the performance of a model F on testing dataset D when trained on T. The above protective noise calculation process can be modeled as a bi-level optimization problem:

$$\begin{aligned} &\max_{\delta} \mathbb{E}_{(x,y)\sim D}\left[L\left(F\left(x;\theta(\delta)\right),y\right)\right] \\ &\text{s.t.} \quad \theta(\delta) = \arg\min_{\theta} \sum_{(x_i,y_i)\in T} L\left(F\left(x_i+\delta_i;\theta\right),y_i\right). \end{aligned} \quad (2)$$

Since defenders cannot access information from the testing set D and directly optimize the Eq. (2), they typically construct a surrogate model F_0. They then generate protective noise targeted at F_0 with the hope of disrupting the intended model F. Subsequent work, inspired by adversarial examples and adversarial training [32–35], has effectively enhanced the transferability of protective noise by training an adversarially robust F_0 [36].

Although unlearnable techniques cannot be directly applied to DFA defense, they offer valuable insights for constructing a universal optimization problem for DFA defense and designing corresponding algorithms.

Against Diffusion Finetuning Attacks: Recent efforts have sought to defend against certain specific DFAs. **AdvDM** [17] targets at trainable parameters. It successfully disrupts TI's imitation performance by identifying adversarial examples for a pretrained DM, leveraging the sensitivity of TI's word embedding, an intermediate trainable parameter. However, AdvDM may be less effective against LR and DB, where attackers directly optimize the backbone UNet parameters. **Glaze** [16] focuses on feature representations. It leverages the feature encoder of DM to optimize and protect images in the feature space, transforming them into a different style. Nonetheless, as this perturbation is limited to style features, it can effectively safeguard artistic styles but not characters or identities. The generalizability challenges faced by existing studies stem from the complex behaviors of attackers, such as heterogeneous trainable parameters and diverse training objectives, which is the primary focus of our work.

3 Threat Model

We consider two parties in our defense: the data owner *(defender)* and the infringer *(attacker)*. The data owner possesses a small set of personal images, which might contain specific characters or art styles. The infringer can launch DFA with any finetuning algorithm and obtain a customized DM that emulates the personal image.

Defender's goals:

▸ **Goal I: Unlearnability.** The shared images cannot be learned by the finetuning algorithm to extract their key visual features, and hence further imitate them.

▸ **Goal II: Usability.** Any defense applied to the images should not compromise their usability. For example, if adding noise, it should be visually imperceptible.

▸ **Goal III: Acceptable computational cost.** As the users who need to protect their images are often individual users with limited computing power. Therefore, the defense algorithm must exhibit computational efficiency.

Defender's capabilities: (○ Unknown ◑ Partially knowable)

◑ **Backbone DM adopt by the attacker:** The Diffusion model used by the attacker has numerous variations based on the training dataset and different architectures [1, 3, 24, 26, 37, 38]. However, most of the existing methods are still not compatible with various finetuning algorithms [37, 38] or are not open-sourced, only providing paid APIs [3, 24]. Adopting these models for attack requires advanced expertise or substantial expenses, which to some extent restricts potential attacks, especially those initiated by ordinary individual users.

Therefore, we assume that *both the defender and the attacker use open-source backbone DMs that are well-adapted for fine-tuning*. We start with the optimistic scenario where they both utilize Stable-Diffusion-V1.5 (***SD-v1.5*** [23]). Then, we gradually relax the attacker's assumptions to explore the generalizability of the defense, allowing the attacker to employ progressively more advanced and divergent architectures, including ***SD-v2*** [26], ***SDXL*** [39], and ***SANA*** [40]. SD-v2 and SD-v1.5 share a similar LDM [26] architecture but achieve better parameters through an improved training process. SDXL introduces a dual-stage network and two text encoders, resulting in larger architectural differences than SD-v1.5. SANA employs a DiT backbone [41] and is trained through rectified flow [42] to achieve state-of-the-art synthesis performance, which is entirely different from SD-v1.5.

○ **Finetuning algorithms adopt by the attacker:** We assume that *the defender is unaware of the specific finetuning algorithm employed by the attacker, such as LR [7], TI [8] or DB [9].* This presents a highly challenging scenario that previous work has not considered. Additionally, practical applications [14, 15] show that using multiple finetuning methods simultaneously is uncommon and typically does not yield significant improvements. Thus, we assume that the attacker uses only one of the three finetuning algorithms at a time.

Attacker's extra capabilities: In addition to the aforementioned foundational settings, we further employ assumptions of attacker's extra capabilities to assess the robustness of the defense mechanisms. We assume that the attacker with sufficient computational resources can execute a *stronger finetuning attack*, such as fully finetuning the entire backbone of DM. Moreover, attackers may employ *adversarial eliminating measures* aimed at neutralizing the protective noise, such as JPEG [43], GRAY [44], BDR [44], SR [45] or TVM [46].

4 Active Defense: Framework and Insight

To develop a more generalizable active defense against DFAs, in this section, we summarize the common behaviors of attackers, thereby conceptualizing active defense as a theoretical bi-level optimization problem to guide the design of defense algorithms. Based on this framework, we analyze the dilemmas of existing defense solutions and gain insight into the design of generalizable defense strategies. This forms the foundation for our universal DFA defense algorithm, `Pretender`, which will be detailed in Sec. 5.

4.1 Bi-level Optimization Problem

Common Attacker Behaviors: The DFAs utilize sophisticated deep neural networks with parameters θ to model a generative probability distribution p_θ, where θ is collectively influenced by multiple sub-network parameters. The typical trainable parameters that are currently of interest to the attacker include the backbone UNet θ_U and the text encoder θ_T used to convert prompts into word embeddings, while there are also parameters that are not trained by default in the current attacks such as the VAE θ_V, which compresses the images into the latent features to support the generation task in latent space. We denote this property as:

$$\theta = \{\theta_U, \theta_T, \ldots, \theta_N\}, \tag{3}$$

where θ_N represents the network parameters that could potentially be utilized for attacks in the future.

We note that, regardless of the specific implementation, the objective of all types of attacks aligns with the standard training goal of generative models: *maximize the likelihood of the target image on* p_θ. Given a specific group of target training data x_0, LR [7] and DB [9] train θ_U to achieve a maximization of $p_{\theta_U}(x_0)$, while TI [8] maximize $p_{\theta_T}(x_0)$ through fine-tuning the θ_T. According to the theoretical principles of generative models, the likelihood maximization means that p_θ could mimic the distribution of authentic training data, leading to the generation of images that visually resemble the target data x_0. Therefore, we universally represent different attacker aims, both current and potential, as aiming to maximize the likelihood of p_θ.

Formulation of Optimization Problems: We denote the personal image data as x. To fulfill the defender's **goal II: usability**, our active defense seeks to protect the image by applying visually imperceptible perturbations δ, such that $||\delta|| < \varepsilon$. This process yields a protected image $x' = x + \delta$.

According to the defender's **goal I: unlearnability**, active defense against DFA can be defined as the following bi-level optimization problem:

$$\delta = \arg\min_{\delta} p_{\theta(\delta)}(x+\delta) \tag{4}$$

$$\text{s.t.} \quad \theta(\delta) = \arg\max_{\theta} p_\theta(x+\delta). \tag{5}$$

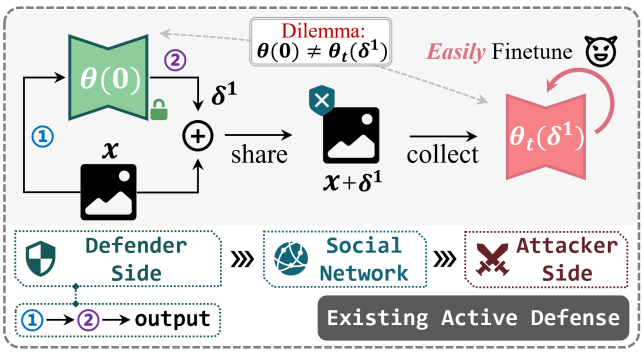

Figure 2: **Optimization dilemma of existing active defense against DFAs.** ① represents image input to the DM. ② refers to gradient back-propagation. θ_t represents the DM finetuned by the attacker, distinct from the surrogate model θ held by the defender.

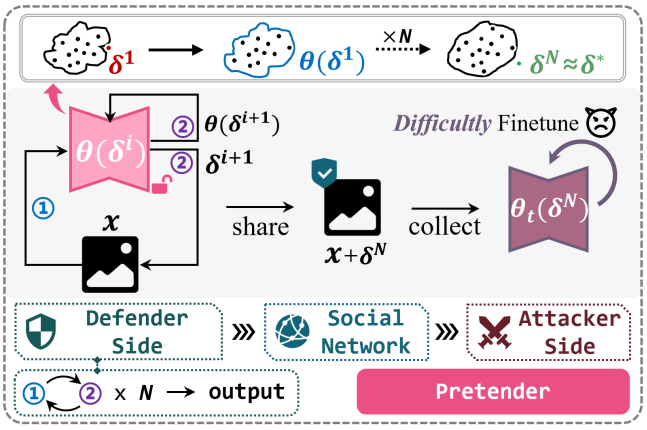

Figure 3: **Defense mechanism of `Pretender`**. The markings are consistent with Fig. 2. The upper part demonstrates the intrinsic principle behind the effectiveness of the defense.

The *outer-level* optimization objective Eq. (4) is to minimize the likelihood of the perturbed image x' under the model $p_{\theta(\delta)}(x)$ trained by the adversary using these images after applying the defense. Here, $\theta(\delta)$ emphasizes the network parameters' dependence on the defense perturbation δ, i.e., the network parameters in this formula are obtained from the training set with defense perturbations. If we minimize $p_{\theta(\delta)}(x')$, it implies that the generative model fails to replicate the visual pattern of x', or more precisely, x. Because $x' = x + \delta$ where the intensity of δ is restricted, they are visually almost the same.

The *inner-level* optimization objective Eq. (5) describes the process by which the attacker is trained. The attacker could launch various DFAs once the protected image x' is obtained. Using the aforementioned maximum likelihood principle, the attacker ultimately receives the network parameters $\theta(\delta)$.

Due to the complexity of directly calculating the likelihood of the generative model $p_\theta(x)$, equivalent proxy objective functions (such as variational lower bounds) are commonly used for optimization. Inspired by [17], we derive and rewrite the bi-level optimization problem as follows (details in appendix B.1):

$$\delta = \arg\max_{\delta} L(\theta(\delta), x+\delta) \tag{6}$$

$$\text{s.t.} \quad \theta(\delta) = \arg\min_{\theta} L(\theta, x+\delta), \tag{7}$$

where $L(\theta, x)$ is a simplified form used to represent the DM-training loss function as introduced in Eq. (1).

4.2 Optimization Dilemma

In the preceding segment, active defense was expounded as a bi-level optimization issue. Nonetheless, it is worth noting that the two levels are intertwined, making it infeasible to compute the protective noise δ through direct optimization.

Existing defense solutions such as `AdvDM` [17] and `Glaze` [16] both employ a *static surrogate model* strategy to approximate optimization objectives. Specifically, they compute protective noise following the Eq. (6) using a pre-trained DM, denoted as $\theta(0)$, whose parameters remain unchanged during computation. To avoid confusion, we will consistently use θ to represent the surrogate model held by the defender and θ_t to denote the DM used by the attacker for DFAs.

However, from the bi-level optimization framework, we derive the following proposition that highlights the insufficiency of current defenses:

Proposition 1. *For any active defense strategies that utilize a static surrogate model* $\theta(0)$ *and optimize through*

$$\delta^1 = \arg\max_{\delta} L(\theta(0), x+\delta), \tag{8}$$

the obtained protective noise δ^1 *is a* ***sub-optimal solution*** *to the bi-level optimization problem (Eq. (6) and Eq. (7)).*

The proof of Proposition 1 is stated in the Appendix B.2. Intuitively, the core dilemma is that $\theta(0)$ fails to reflect the adversarial optimization behavior of attackers (Eq. (7)) after they acquire protected images, as shown in Fig. 2.

Based on aforementioned inferences, we arrive at a core idea that could potentially overcome the current dilemma and achieve the ultimate bi-level optimization goal: *Discovering a* δ^* *that obstructs attackers from minimizing the DM-loss (7), facilitated by a* ***dynamically*** *updated surrogate model.*

5 Pretender

5.1 Technical Motivation

The research on adversarial examples and adversarial training [32–35], as well as robust surrogate model for unlearnable

sample [36], can provide valuable insights for finding δ^*. As illustrated in Fig. 3, the uneven distribution of features space or rough decision boundary (specific for classification models) in deep neural networks leads to adversarial examples, which cause outlying data points (such as δ^1 in Eq. (8)) to be identified very easily. Retraining the model with adversarial examples is equivalent to fixing the vulnerabilities, and it leads to a smoother decision boundary of the model. Assume that we repeat this process several times, i.e.:

$$\theta(0) \to \delta^1 \to \theta(\delta^1) \to \delta^2 \to \theta(\delta^2) \to ... \to \delta^N, \tag{9}$$

and disturb the "repaired" model with adversarial attacks subsequently. It is likely to cause a greater perturbation (compared to δ^1) to the original sample x in the feature space by the successful discovery of δ^N, rendering the attacker's DM θ_t more challenging to be trained on them. Thus, we conclude that δ^N is a useful approximation of δ^*. We further analyze the optimality of this dynamic approach in Appendix B.3.

5.2 Defense Algorithm

We introduce the `Pretender` defense algorithm which employs a *dynamic surrogate model* strategy outlined in Eq. (9). Commencing from a pre-trained DM $\theta(0)$, the algorithm iteratively alternates between computing protective noise and updating network parameters in an adversarial fashion. This process culminates in the final protective noise δ^N.

The protective noise computation process is derived from the FGSM attack [47] and can be expressed as:

$$\delta^{i+1} = \delta^i + \alpha \cdot \text{sign}\left(\nabla_x L(\theta(\delta^i), x+\delta^i)\right). \tag{10}$$

The surrogate model update adheres to the gradient descent formula as follows:

$$\theta(\delta^{i+1}) = \theta(\delta^i) - \beta \cdot \left(\nabla_\theta L(\theta(\delta^i), x+\delta^{i+1})\right). \tag{11}$$

To enhance the efficacy and computational efficiency of model update, we focus on training the crucial UNet parameters θ_U in DM, utilizing Low-Rank Adaptation techniques [30] for acceleration. Specifically, we denote the parameter update matrix as $W = \theta(\delta^{i+1}) - \theta(\delta^i)$, where $W \in \mathbb{R}^{d\times k}$. We then employ a low-rank decomposition $W = BA$, where $B \in \mathbb{R}^{d\times r}, A \in \mathbb{R}^{r\times k}$ with the rank $r \ll \min(d,k)$.

However, the aforementioned iterative adversarial training poses significant challenges in determining the optimal number of training epochs for both noise computation and model update, accompanied by considerable computational overhead [32, 33, 47]. Considering the defender's **goal III: acceptable computational cost**, we propose the *simultaneous gradient back-propagation* (SGBP) strategy to decrease computational overhead.

Simultaneous Gradient Back-Propagation: It has been observed that the most time-consuming part of the two equations (10) and (11) is the gradient calculation. Currently used open-source deep learning frameworks can calculate gradients with respect to images and network parameters simultaneously. Meanwhile, by calculating the gradient with respect to images following one forward pass and then computing the network parameter gradient after another forward pass, the total time required greatly exceeds that necessary to calculate the gradients with respect to both images and network parameters simultaneously after one forward pass.

Therefore, we approximate the gradients calculation in Eq. (11) as

$$\nabla_\theta L(\theta(\delta^i), x+\delta^{i+1}) \approx \nabla_\theta L(\theta(\delta^i), x+\delta^i).$$

We can then perform a forward pass that calculates $L(\theta(\delta^i), x+\delta^i)$, and then update both the defense perturbation δ and the network parameters $\theta(\delta)$ by running the gradient back-propagation only once. The iterative process enables the network to learn about $\delta^i, i \in (1, N-1)$ during parameter updates and approximately achieve our final bi-level optimization goals. The effectiveness and time efficiency of this optimization strategy would be demonstrated by experiments.

We illustrate the `Pretender` defense in Fig. 3 and describe the detailed procedures in Algorithm 1.

Algorithm 1: Pretender

Input: Personal image x, Diffusion parameter θ, disturb strength α, disturb budget ε, training strength β, optimization step N

Output: Protected image x'

$x^0, \theta^0, \delta^0 \leftarrow x, \theta, 0$;
for $i = 1$ **to** N **do**
  $t \leftarrow$ randomSample();
  $L^i \leftarrow L(\theta^i, x^i, t)$ `/* DM-loss */`;
  $G_x^i, G_\theta^i \leftarrow \nabla_{x,\theta} L^i$ `/* SGBP */`;
  $\delta^i \leftarrow \delta^{i-1} + \alpha \cdot \text{sign}(G_x^i)$;
  $\delta^i \leftarrow \text{clip}(\delta^i, \varepsilon)$;
  $x^i \leftarrow x^0 + \delta^i$;
  $\theta^i \leftarrow \theta^{i-1} - \beta \cdot G_\theta^i$ `/* Update` θ_U `*/`;
end
return $x' \leftarrow x^0 + \delta^N$;

6 Evaluation Setup

6.1 Overview

We conduct comprehensive evaluations of the proposed `Pretender` framework through a broad array of task scenarios, diverse metrics, and comparisons with representative baselines. This section will detail the experimental setup, including the datasets utilized and the metrics employed. In

subsequent sections, we will present the four main components of our experimental evaluation: effectiveness (Sec. 7), robustness and transferability (Sec. 8), usability and time efficiency (Sec. 9), and real-world performance (Sec. 10). These evaluations collectively demonstrate how our proposed framework achieves the defender's goals in active defense tasks.

6.2 Tasks and Datasets

Our evaluation encompasses two of the most prevalent applications of the Diffusion Model (DM): Text-to-Image (T2I) and Image-to-Image (I2I). In the T2I scenario, we assess the defensive efficacy for the task of *Character Reproduction*. In the I2I application, we evaluate the protective capabilities against *Style-Imitation*.

To faithfully simulate the security risk scenario where an adversary maliciously uses a small amount of data from individual users for fine-tuning pre-trained DMs, we manually collect a **minimal** set of **visually similar** images (e.g., 5 or 10 images) in all tasks, which forms a basic experimental group. The process is repeated to create multiple groups, covering a wide variety of image categories.

For each basic experimental group, the DMs are finetuned from four separate sources: original images (No-Protect), images protected with `AdvDM` [17], `Glaze` [16] and protected with `Pretender` **(ours)**. After finetuning, we generated 100 images using each DM for further evaluation across various metrics.

6.2.1 Character-Reproduction

In this T2I task, a flexible textual description prompt is used to generate specific backgrounds, postures, and shooting angles for certain characters. Additionally, given the sensitivity of facial images, generating visual content for specific individuals (deepfakes) and the associated protection mechanisms are also worth exploring. Therefore, we have utilized the two datasets listed below:

① **LSUN [48]**. It consists of diverse animals or objects, such as horses, cats, cars, airplanes, and more. We selected **10** categories and manually chose **5** sets of **5** visually similar images from each category, totaling **50** groups.

② **RAVDESS [49]**. It comprises high-resolution videos of multiple actors hired to record videos of various facial expressions. We selected **20** actors from each of whom we cropped **10** facial images, resulting in a total of **20** distinct groups. We attempted to select a wide range of facial expressions to simulate collecting private photos from social media or public photos of celebrities.

6.2.2 Style-Imitation

In this I2I task, the infringers imitate the target *style* by fine-tuning a DM with stylistically coherent artworks. A *content* image, laden with structural details like a landscape or draft, is fed into the DM. The principal aim is to leverage the DM to enrich the image's textural data, thereby effectively imbuing the input *content* image with the desired *style*. To simulate various distinctive artistic styles, we use the following dataset:

③ **WikiArt** [50]. It spans multiple artistic genres, including but not limited to abstract art, pointillism and sketch. We selected **10** different artistic genres and chose **5** sets of **5** similar artworks within each genre, totaling **50** groups.

The *content* images, onto which a specific imitated style was transferred, were selected from the **Pexels** [51]. The dataset contains authorized photographs of various categories, such as landscapes and portraits of people.

6.3 Metircs

We employ **automated evaluation metrics** that deliver objective and quantitative assessments through computational models. Meanwhile, we utilize **human evaluation metrics** derived from user studies, where individuals assess outputs based on perceptual criteria, offering qualitative insights into the user experience. The adoption of these holistic metrics significantly enhances the validity of the results, ensuring a meticulous evaluation in our study.

The goal of aforementioned tasks is for the DM to produce images that deviate from the original distribution after fine-tuning on protected images, thereby achieving the desired protective effect.To assess this deviation, our metrics primarily focus on two aspects: the *visual quality* and the *semantic similarity* of the generated images. Overview of the metrics adopted in each dataset is exhibited in Tab. 1.

6.3.1 Automated evaluation metrics

For the assessment of *visual quality*, we utilize Fréchet Inception Distance (**FID**) [52] and Precision (**Prec.**) [53] for LSUN evaluations. Both metrics evaluate the distributional similarity between generated images and original images. The Structural Similarity Index Measure (**SSIM**) [54] is additionally employed for RAVDESS dataset.

For the *semantic similarity* evaluations, due to the lack of universally applicable metrics in previous studies, we have designed two metrics named BLIP-Matching-Score (**BMS**) and BLIP-Similarity-Score (**BSS**) for LSUN dataset. Specifically, we initially employ BLIP [55] to caption images in each training group, extracting a common caption like 'a yellow sports car'. We then calculate the image-text matching score and cosine similarity in BLIP's feature space for each generated image and the caption. Additionally, **ArcFace** [56] is utilized to assess the similarity between synthesized and original human faces in the RAVDESS dataset, which projects facial features into a hyperspherical space, facilitating the measurement via cosine similarity. Details are illustrated in appendix A.1.

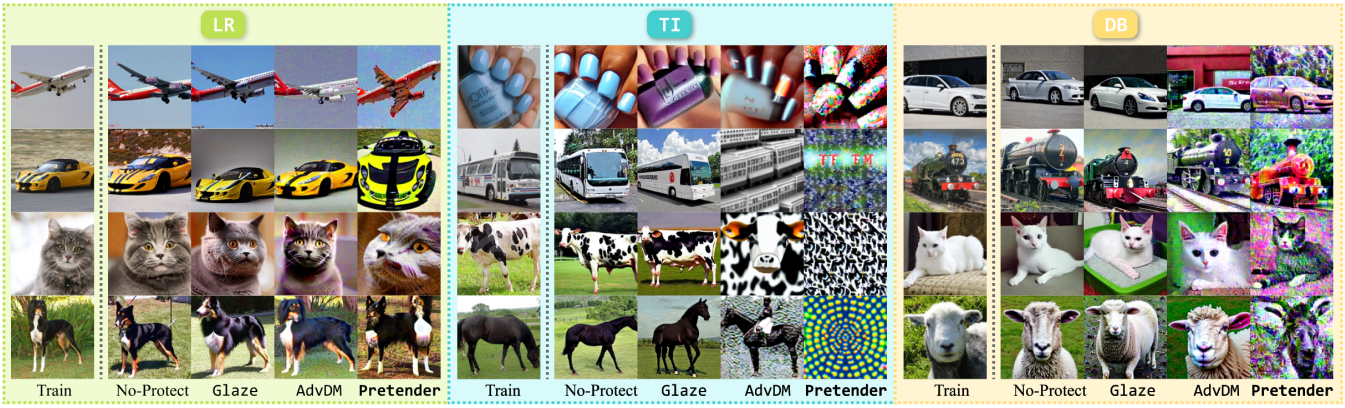

Figure 4: **Qualitative evaluations of image protection in character-reproduction task based on LSUN [48] dataset.** In each group, the leftmost column shows an training image, while the right columns show the images generated by the DFAs.

Metric→ Dataset↓	Automated Eva.						Human Eva.	
	FID	**Prec.**	**SSIM**	**ArcFace**	**BMM**†	**BMS**†	$\boldsymbol{H}_{\mathrm{Q}}$†	$\boldsymbol{H}_{\mathrm{S}}$†
Character-Reproduction								
① **LSUN**	●	●			■	■	●	■
② **RAVDESS**	●		●	■			●	■
Style-Imitation								
③ **WikiArt**							●	■

Table 1: **Tasks and datasets along with the adopted metrics.** ● represents *visual quality* metrics, and ■ represents *semantic similarity* metrics. † represents the metric newly proposed in our study. RAVDESS following the settings of [49], and WikiArt following the setting of [16, 17].

6.3.2 Human evaluation metrics

We propose a human-evaluated visual quality score, denoted as $\boldsymbol{H}_{\mathrm{Q}}$, and a human-evaluated semantic similarity score, referred to as $\boldsymbol{H}_{\mathrm{S}}$. We initially select one image randomly generated by different DMs (finetuned by four differently protected image sources) within the same experimental group to form a test set. Each test set comprises one image from each of the four settings: No-Protect, `Glaze`, `AdvDM` and `Pretender`.

Subsequently, we recruited 50 participants, each required to complete 100 test sets. In each set, they voted on four images, selecting the one with the *poorest visual quality* and the one with the *lowest semantic similarity*, indicative of superior protection. Finally, we calculated the proportion of votes for each of the four settings. The proportion of votes for the image with the poorest visual quality was used as the protection method's H_{Q}, expressed as a percentage. Similarly, the proportion for the image with the lowest semantic similarity was recorded as H_{S}. Higher H_{Q} and H_{S} signify more effective protection.

6.4 Implementation Details

DM-backbone: In effectiveness and robustness evaluations (Sec. 7 & Sec. 8.2), both defender and attacker adopt SD-v1.5. In transferability evaluations (Sec. 8.3), the defender uses SD-v1.5 while the attacker utilizes SD-v2, SDXL and SANA.

DFA algorithm: In all experiments with LSUN and WikiArt, we employ LoRA (**LR**) [7], Textual-Inversion (**TI**) [8], and DreamBooth (**DB**) [9]. For RAVDESS, we only adopt **LR**, as it is the only one among the three DFAs that effectively mimics facial images, based on experiences from the open-source community. All DFAs are implemented by the `diffuser` framework.

For more comprehensive information regarding the implementation details of the finetuning algorithms and defense algorithms, please refer to the appendix A.2.

7 Effectiveness Evaluation

7.1 Defend Character-Reproduction

Initially, we evaluate the protective efficacy of our proposed `Pretender` for commonly depicted character images, conducting both quantitative and qualitative assessments based on the LSUN dataset. Quantitative evaluations include automated metrics displayed in Tab. 2, with detailed performance across different categories presented in Fig. 6. Additionally, the results of human-evaluated metrics are shown in Fig. 5. Concurrently, we illustrate and compare the visual outcomes of images generated by various protection algorithms as the qualitative evaluation, which are exhibited in Fig. 4.

Overall, our protective algorithm demonstrates remarkable advantages in safeguarding commonly depicted character images under various DFA settings, effectively disrupting the synthetic images' both *visual quality* and *semantic similarity*. `Glaze`, designed exclusively to safeguard artistic styles, fails to prevent DFAs from extracting character information. Quantitative metrics reveal that its performance is comparable to unprotected settings. `AdvDM`, engineered specifically to counteract TI, achieves competitive performance in disrupting TI's visual quality, attaining the best FID and Prec. performances. However, its protective capabilities are inadequate across other DFA settings. Our proposed `Pretender` comprehensively disrupts image quality

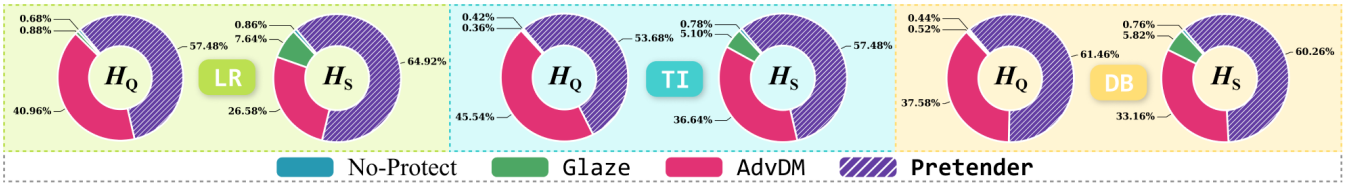

Figure 5: **Human evaluation for the protection effectiveness on LSUN dataset.**

Attack method → / Defense method ↓	LR				TI				DB			
	FID↑	**Prec.↓**	**BMS↓**	**BSS↓**	**FID↑**	**Prec.↓**	**BMS↓**	**BSS↓**	**FID↑**	**Prec.↓**	**BMS↓**	**BSS↓**
No-Protect	59.244	0.344	0.812 / 0.951	0.422	60.329	0.304	0.740 / 0.943	0.432	58.080	0.524	0.795 / 0.921	0.414
Glaze [16]	58.488	0.357	0.838 / 1.000	0.431	77.569	0.248	0.675 / 0.920	0.428	62.527	0.463	0.785 / 0.951	0.429
AdvDM [17]	86.358	0.372	0.779 / 0.892	0.400	**229.200**	**0.056**	0.161 / 0.003	0.272	109.328	0.267	0.741 / 0.898	0.406
Pretender**(Ours)**	**92.566**	**0.316**	**0.766 / 0.880**	**0.383**	182.607	0.076	**0.124 / 0.002**	**0.248**	**155.622**	**0.100**	**0.603 / 0.708**	**0.380**

Table 2: **Quantitative evaluations of image protection in character-reproduction task based on LSUN dataset.** For **BMS**, we report **(mean / median)**. Higher FID indicates better protection; lower values are preferable for other metrics.

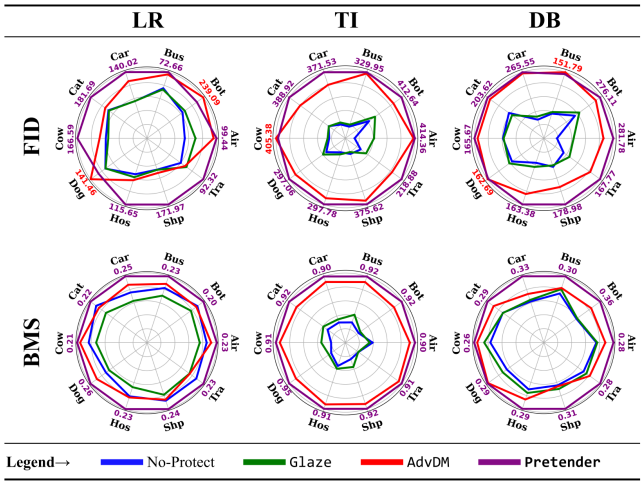

Figure 6: **Quantitative evaluations on LSUN (class-wise).** We selected one visual quality metric (FID) and one semantic similarity metric (BMS) for demonstration, wherein we present *'1-BMS'* for comparison. Each vertex in the radar chart represents the performance across different categories, with larger values indicating better protection effectiveness.

across three different DFA settings and achieves optimal protection outcomes in all semantic similarity safeguard metrics (BMS and BSS). The visually protective effects displayed in Fig. 4, along with the subjective evaluation results presented in Fig. 5, further corroborate the aforementioned conclusions.

We further focus on the protective efficacy for facial images based on RAVDESS dataset. Quantitative evaluations include both automated and human evaluation metrics, displayed in Tab. 3, while visual quality assessments are shown in Fig. 7. Comprehensive experimental results indicate that **our defense algorithm not only disrupts the quality of synthetic images but also significantly impedes DFA from reproducing *facial identities*, outperforming existing protections.**

Metric→ / Defense↓	Automated Evaluation			Human Evaluation	
	FID↑	**SSIM↓**	**ArcF.↓**	H_Q↑	H_S↑
No-Protect	124.36	0.45	0.72	0.55	4.15
Glaze [16]	131.29	0.46	0.70	0.50	3.40
AdvDM [17]	198.58	0.49	0.67	39.70	22.70
Pretender	**207.82**	**0.40**	**0.55**	**59.25**	**69.75**

Table 3: **Quantitative evaluations of human facial image protection.** ArcF. refers to ArcFace metric.

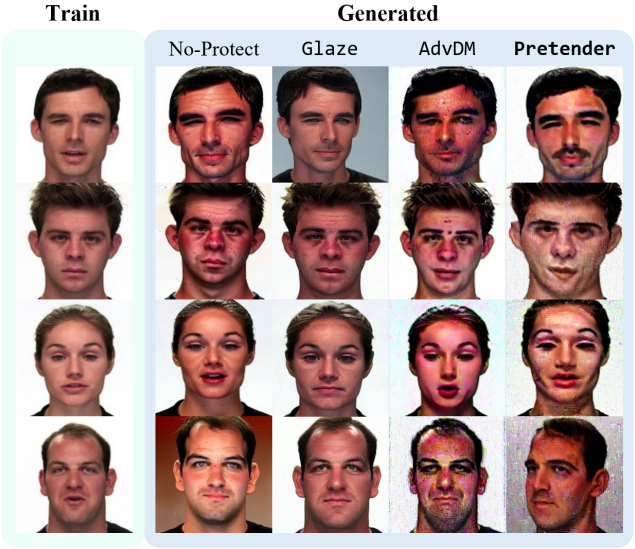

Figure 7: **Qualitative evaluations of human facial image protection.** The leftmost column shows an image from the training set, while the right columns show the images generated by the trained DM with LR finetuning algorithm. AdvDM and Glaze protection still enables the DFA to extract most of the facial identity information. Only Pretender could successfully affect the identity stealing of DM by altering facial features such as the position and shape of the facial organs.

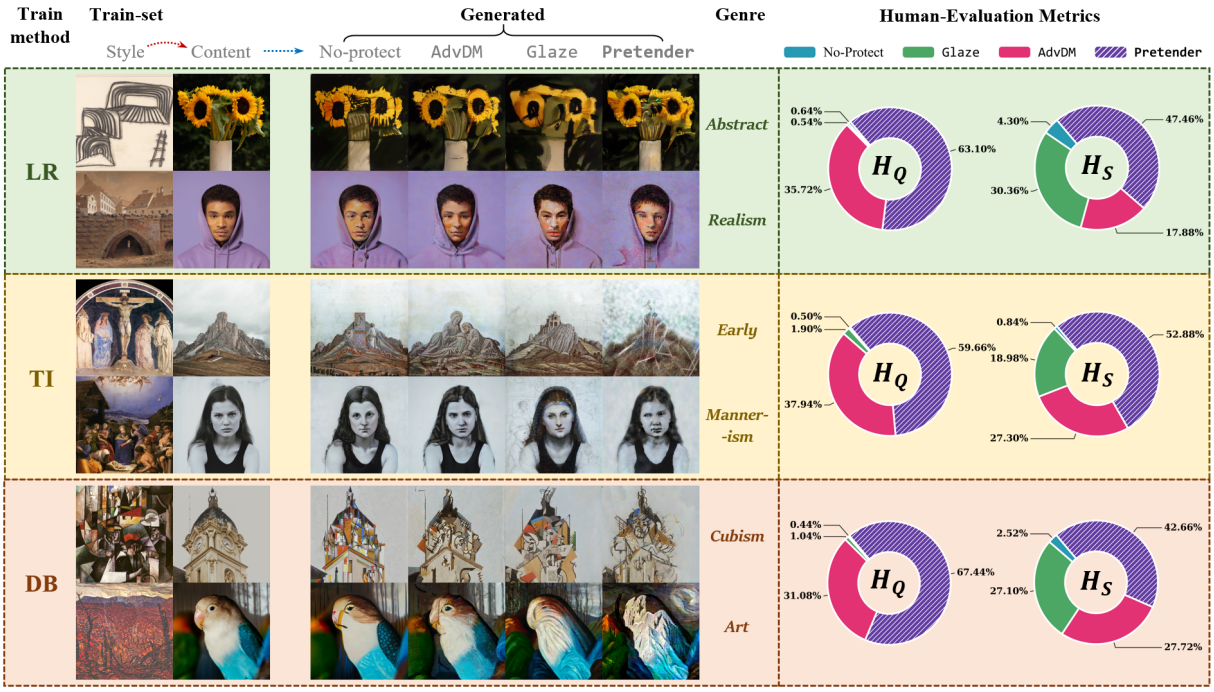

Figure 8: **Qualitative evaluations of image protection in style-imitation task based on WikiArt [50] dataset.**

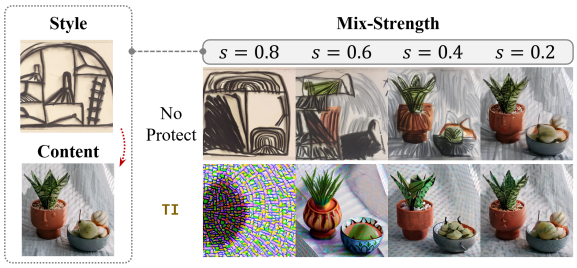

Figure 9: **Image protection effect on style-imitation under different mix-strength settings**.

7.2 Defend Style-Imitation

We evaluated the protective effectiveness for stylized images and artistic works within a widely-used Image-to-Image (I2I) attack scenario, based on the WikiArt dataset. Visual quality assessments and human-evaluated quantitative evaluation results are both presented in Fig. 8.

Our defense algorithm more effectively resists style imitation attempts by various DFAs. `AdvDM` provides only limited protection except for TI. `Glaze` can achieve a certain degree of protection effect because its algorithm aims to disturb an artistic image to be identified as another artistic style within the feature space. In comparison, `Pretender` can more effectively interfere with the DFA's imitation of the style. For instance, in the *DB-cubism* group, the DM is unable to reproduce color block mosaics present in the style image, and the colors tend to be simplified to a single tone. More detailed results are presented in the supplementary files.

Furthermore, noting that the visual quality of synthesized images in this I2I task is not as significantly disrupted as in several T2I scenarios (as in Fig. 4 and Fig. 7), we delved deeper into the impact of mix-strength on the defense against style imitation. Specifically, mix-strength, denoted as $s \in [0, 1]$, regulates the intensity of style transfer conducted by the DFA. s closer to 1 signifies that the generated image contains more of the target style [28]. We demonstrate the effects of various s using the `Pretender` on TI in Fig. 9, with the complete results for multiple DFAs presented in Fig. 17 the appendix.

We discovered that **I2I tasks require the defense algorithm to more extensively disrupt the semantics of synthesized images, an area where our defense proves particularly adept.** As illustrated in Fig. 9, infringers typically opt for a moderate mix-strength, such as $s = 0.4 \sim 0.6$, to achieve a balance between the content and style of the images. At this level, the visual quality disturbances caused by the defense algorithm are mitigated, while more of the semantic information (i.e., the style extracted by the DM) is preserved. Take the `TI` results for instance, where a decrease in s gradually transforms the image from a noisy pattern to a meaningful content with less noise, but the generated images failed to imitate the target style, achieving a successful defense.

> ***Takeaway 1.*** `Pretender` offers effective image protection against various finetuning attacks, disrupting both the visual quality and semantic similarity of synthesized images by DMs across multiple T2I and I2I tasks. Its effectiveness is validated by a broad range of objective quantitative metrics and subjective measures.

DFAs → Adv. Meas. ↓	LR				TI				DB			
	FID↑	**Prec.↓**	**BMS↓**	**BSS↓**	**FID↑**	**Prec.↓**	**BMS↓**	**BSS↓**	**FID↑**	**Prec.↓**	**BMS↓**	**BSS↓**
No-Protect	59.244	0.344	0.812 / 0.951	0.422	60.329	0.304	0.740 / 0.943	0.432	58.080	0.524	0.795 / 0.921	0.414
Pretender-Ori												
w/o adv	92.566	0.316	0.766 / 0.880	0.383	249.211	0.036	0.081 / 0.002	0.250	143.371	0.132	0.697 / 0.855	0.397
JPEG [43]	69.178	0.338	0.773 / 0.881	0.416	133.735	0.117	0.626 / 0.839	0.403	77.889	0.458	0.764 / 0.938	0.412
BDR [44]	88.770	0.136	0.783 / 0.952	0.417	141.295	0.200	0.256 / 0.012	0.301	182.920	0.196	0.700 / 0.787	0.398
GRAY [44]	97.446	0.040	0.689 / 0.830	0.402	161.146	0.252	0.524 / 0.597	0.364	160.803	0.237	0.652 / 0.759	0.390
SR [45]	67.219	0.346	0.722 / 0.844	0.417	136.916	0.121	0.601 / 0.743	0.408	85.985	0.424	0.639 / 0.771	0.403
TVM [46]	102.628	0.312	0.787 / 0.909	0.397	170.659	0.103	0.542 / 0.619	0.382	83.915	0.308	0.753 / 0.913	0.408
Average	85.048	0.234	0.751 / 0.883	0.410	148.750	0.159	0.510 / 0.562	0.372	118.302	0.325	0.702 / 0.834	0.402
Pretender-Aug												
w/o adv	87.452	0.116	0.791 / 0.927	0.406	188.919	0.042	0.281 / 0.010	0.320	172.601	0.124	0.710 / 0.817	0.401
JPEG [43]	81.000	0.108	0.751 / 0.863	0.396	150.471	0.072	0.433 / 0.355	0.358	148.387	0.112	0.655 / 0.766	0.398
BDR [44]	89.045	0.108	0.758 / 0.906	0.401	196.438	0.116	0.233 / 0.005	0.298	156.079	0.096	0.707 / 0.763	0.383
GRAY [44]	110.682	0.048	0.678 / 0.846	0.408	169.177	0.164	0.407 / 0.261	0.334	159.247	0.201	0.662 / 0.765	0.391
SR [45]	69.862	0.188	0.730 / 0.835	0.411	115.204	0.156	0.658 / 0.739	0.405	91.932	0.064	0.627 / 0.809	0.408
TVM [46]	99.780	0.140	0.794 / 0.908	0.404	149.350	0.064	0.510 / 0.515	0.385	82.968	0.236	0.739 / 0.902	0.405
Average	90.074	0.118	0.742 / 0.872	0.404	156.128	0.114	0.448 / 0.375	0.356	127.723	0.142	0.678 / 0.801	0.397
Improvement	**15.1%**	**50.9%**	**18.9% / 16.5%**	**15.4%**	**3.9%**	**16.5%**	**9.4% / 19.9%**	**8.6%**	**11.0%**	**46.6%**	**24.2% / 49.2%**	**29.6%**

Table 4: **Robustness to Adversarial Measures.** `Pretender-Aug` a rehearsal-based augmentation approach described in Sec. 8.2. The improvement is calculated through (`Aug` - `Ori`) / ('w/o adv' - 'No-Protect').

8 Robustness and Transferability

8.1 Overview

This section primarily evaluates the more challenging threat model with enhanced attacker's capabilities. We assessed the following three aspects for `Pretender`: ① **Robustness to stronger DFA:** the attacker has an adequate attack budget to finetune the whole DM backbone network. ② **Robustness to adversarial eliminating measures:** the attacker will neutralize the protective noise through preprocessing the images. ③ **Transferability to different DM:** the attacker may adopt a more advanced DM, differing from that used by the defender. The evaluations are primarily based on the character-reproduction task using the LSUN dataset.

8.2 Robustness Evaluation

Firstly, we assumed that attackers can use more computational resources to directly finetune the entire backbone network, the Unet θ_U. We labeled this strong attack as **"UNet"** and evaluated the protective effects of various defense mechanisms under this setting (Tab. 5). Results indicate that existing protective algorithms struggle to withstand stronger DFAs, yet `Pretender` **effectively disrupt both the synthetic image quality and semantics of stronger DFAs**.

Attack method → Defense setting ↓	UNet			
	FID↑	**Prec.↓**	**BMS↓**	**BSS↓**
No-Protect	55.418	0.338	0.794 / 0.917	0.4337
`AdvDM` [17]	74.629	0.343	0.812 / 0.931	0.4312
`Glaze` [16]	63.505	0.329	0.840 / 0.990	0.4292
`Pretender`	**97.654**	**0.307**	**0.738 / 0.872**	**0.4193**

Table 5: **Robustness to Stronger DFA**

We evaluated the impact of a broad range of image preprocessing techniques on defense performance, including JPEG compression (**JPEG**) [43], Bit-Depth Reduction (**BDR**, with 6-bit quantization) [44], grayscale processing (**GRAY**) [44], Super-Resolution (**SR**) [45], and Total Variation Minimization (**TVM**) [46]. The `Pretender-Ori` setting in Tab. 4 display the evaluation results, indicating that **adversarial measures can somewhat diminish the effectiveness of our protection, however, our defense retains robustness in certain settings.** For example, none of the adversarial measures can completely eliminate the impact of our protection when adopting TI.

We further enhance the robustness of `Pretender` through a rehearsal-based approach, which involves randomly applying noise-removal operations during the training of protective noise, thereby forcing the generated noise to become more robust. Specifically, we randomly introduce **JPEG** and **BDR** before the image is input into the network (before Line 4 in Algorithm 1), corresponding to the `Pretender-Aug` settings in Tab. 4. Experimental results show that `Pretender-Aug` achieves performance similar to `Pretender-Ori`, while demonstrating an overall improvement in robustness.

8.3 Transferability Evaluation

We validated the transferability of `Pretender` using only SD-v1.5 [26] against more advanced DMs. We employed progressively more difficult attacker settings, including SD-v2 [26], SDXL [39], and SANA [40]. SD-v2 shares the most similar architecture, but achieves better parameters through an improved training process. SDXL introduces a dual-stage network and two text encoders, resulting in architectural differences. SANA employs a completely different DiT backbone.

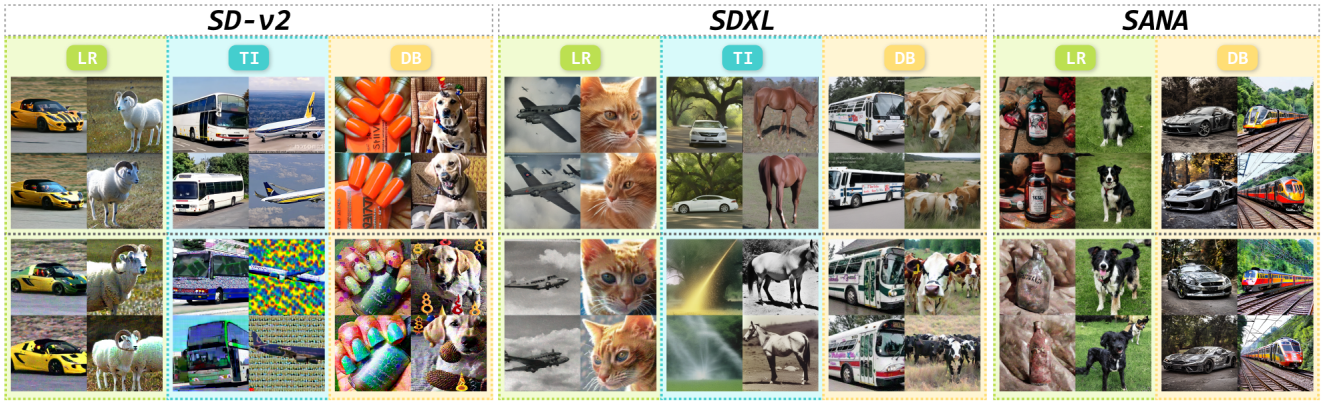

Figure 10: **Visual quality demonstration when Pretender defends against more advanced DMs.** (Upper part: No-protect)

DFAs	FID↑	Prec.↓	BMS↓	BSS↓
		SD-v1.5 [26]		
LR	59.244	0.344	0.812 / 0.951	0.422
	92.566	0.316	0.766 / 0.880	0.383
TI	60.329	0.304	0.740 / 0.943	0.432
	182.607	0.076	0.124 / 0.002	0.248
DB	58.080	0.524	0.795 / 0.921	0.414
	155.622	0.100	0.603 / 0.708	0.380
		SD-v2 [26]		
LR	52.460	0.357	0.831 / 1.000	0.427
	118.384	0.287	0.790 / 0.925	0.408
TI	52.184	0.383	0.807 / 1.000	0.434
	178.864	0.082	0.503 / 0.544	0.348
DB	47.418	0.535	0.856 / 1.000	0.435
	127.256	0.232	0.740 / 0.873	0.417
		SDXL [39]		
LR	53.537	0.244	0.810 / 1.000	0.428
	65.628	0.096	0.793 / 0.932	0.417
TI	64.669	0.202	0.721 / 0.953	0.416
	76.638	0.144	0.605 / 0.763	0.398
DB	53.921	0.236	0.839 / 1.000	0.441
	69.974	0.096	0.806 / 0.947	0.424
		SANA [40]		
LR	65.393	0.8664	0.759 / 1.000	0.429
	69.186	0.8152	0.689 / 0.835	0.414
DB	82.035	0.7931	0.425 / 0.241	0.387
	85.621	0.6732	0.409 / 0.212	0.360

Table 6: **Transferability to DFAs from different DMs.** (GRAY-font: No-protect, BLACK-font: `Pretender`)

The evaluation results are presented in Tab. 6 and Fig. 10. The proposed defense can **effectively transfer to more advanced DMs**. Additionally, we observed that greater architectural differences lead to a reduction in the defense's effectiveness, especially in terms of visual quality distortion. Nevertheless, `Pretender` can still induce semantic distortion in certain images.

Takeaway 2. `Pretender` exhibits robustness against some adversarial measures, which could be further enhanced through a rehearsal-based approach. It also demonstrates transferability to more advanced DMs.

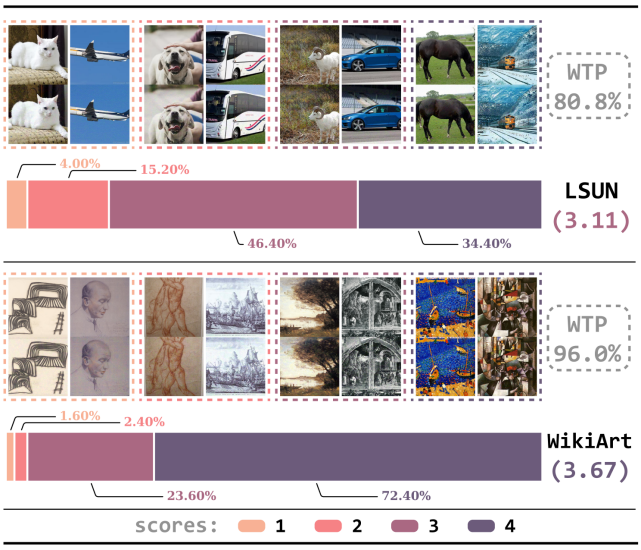

Figure 11: **User ratings for image usability.** Higher scores indicate that users are less aware of the protective measures applied by `Pretender` in mages, signifying greater usability. We presented samples with corresponding scores in different colored boxes. `WTP` refers to the 'willingness to post' ratio, which equals the proportion of scores rated as 3 or 4.

9 Usability and Time Efficiency

9.1 Usability Evaluation

Settings: we conducted a usability assessment based on a user study. 10 participants were recruited to assessed 50 images (500 images in total) at a high resolution of 512×512, all from LSUN/Wikiart and protected by `Pretender`. They rated on a scale from 4 to 1 which includes: (4) Imperceptible-noise; (3) slightly-perceptible-noise but acceptable; (2) perceptible-noise but conditionally-acceptable; (1) completely-unacceptable-noise. During the rating participants were also shown the original version of the protected images for comparison. Additionally, participants were in-

Defense	Time (min)↓	FID↑	Prec↓	BMS↓	BSS↓
Glaze [16]	3.58 ± 0.31	66.19	0.36	0.77	0.43
AdvDM [17]	**1.14 ± 0.08**	141.63	0.23	0.56	0.36
Pretender **(Ours)**	2.29 ± 0.12	**161.72**	**0.16**	**0.51**	**0.34**

Table 7: **Comparison of time efficiency with existing defense algorithms.**

structed to give a score of 3 or 4 if they would be willing to post the protected image instead of the original version. Therefore, we calculated the total proportion of 3 and 4 as the willingness metric.

Results: We exhibit the results in Fig. 11. The assessment confirms that Pretender **does not affect the usability of the images.** We conclude that the scores are correlated with the characteristics of the original images. Simpler structures and lighter colors (e.g., clean sky backgrounds or white objects) make protective noise more perceivable. As a result, object images (LSUN) generally receive lower scores compared to artistic images (WikiArt). Besides, we also investigate the trade-off between usability and defense effectiveness in appendix A.3.

9.2 Time Efficiency Evaluation

We first compared the time efficiency of our defense algorithm with existing solutions. Using the LSUN dataset, we measured the time taken to apply protection to each experimental group (comprising 5 images) and reported the mean and standard deviation in minutes. In addition, we compile the average scores of each defense under various DFAs, presenting them collectively in Tab. 7.

Our algorithm exhibits moderate time overhead while achieving the best effectiveness compared to existing defenses. Pretender takes approximately twice as long as AdvDM because we utilize the official optimization step setting (100 steps vs 40 steps). The time consumption of Glaze shows relatively large fluctuations, due to the inherent uncertainty in its search process. Overall, we believe that a time expenditure of 1 to 2 minutes to protect a set of personal images *is an acceptable overhead for individual users*.

We subsequently evaluate the effectiveness *Simultaneous Gradient Back-Propagation* (SGBP) strategy proposed in Pretender and demonstrate how it optimizes the utilization of computational resources.

Settings: For a typical iterative optimization strategy, which updates the image for a steps while keeping the network unchanged, and then trains the network for b steps while keeping the image unchanged, we denote this as O_aI_b. Conversely, SGBP optimizes both the image and the model simultaneously during a single gradient backpropagation, which is denoted as $(OI)_1$. We compare its performance with four strategies: $(OI)_1$, O_1I_1, O_5I_5, and $O_{10}I_{10}$. We define one iteration as one complete update of both the image and network

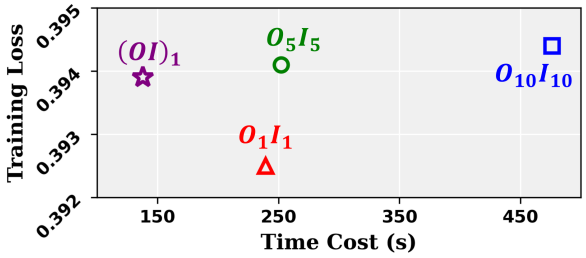

Figure 12: **Comparison between the proposed SGBP strategy $(OI)_1$ and several possible iterative optimization strategies.** A larger loss represents a better protective effect.

parameters. Then we set the number of iterations for $(OI)_1$, O_1I_1 strategies to 100, and set it for O_5I_5, $O_{10}I_{10}$ to 20. We recorded the time overhead for generating protection using different strategies, and then trained the same model for 150 steps on various protected images, noting the training loss at termination. A larger loss represents a better protective effect.

Results: Fig. 12 displays the comparison results. Detailed comparison of the training processes can be found in Fig. 16 in the appendix. **SGBP significantly optimizes the time efficiency while also delivering superior defensive effects.** We recorded the training loss for unprotected images as a reference, which resulted in a loss of 0.197. The effectiveness of $(OI)_1$ surpasses that of the approximately equivalent O_1I_1 strategy, but SGBP also saves nearly half the time in terms of computational overhead (137.8s vs 293.3s). When compared to the O_5I_5 and $O_{10}I_{10}$ strategies, SGBP offers a more pronounced advantage in terms of time efficiency, while the performance drop is negligible (less than 0.001).

> ***Takeaway 3.*** Pretender does not compromise the usability of images. In addition, the SGBP strategy enables Pretender to apply protection in a time-efficient manner, facilitating its deployment and application.

10 Real-World Performance

We evaluate Pretender against a real-world online AI-creation system, scenario [57]. It is a web application that allows users to upload a minimum of five images with similar characteristics to serve as a dataset. Subsequently, it trains a model online and provides users with a callable endpoint for image synthesis. Notably, although it is SD-based, its initial model parameters and architecture are closed-source.

We conduct both the character reproduction and the style imitation tasks on scenario. We only adopt the dataset setting described in Sec. 6.2, and ensure that all remaining settings adhere to the default configurations of scenario. Partial results are presented in Fig. 13, and the evaluation demonstrates the efficacy of our approach in real-world systems. Images protected by Pretender can prevent AI-creation software from appropriating their specific characters or styles.

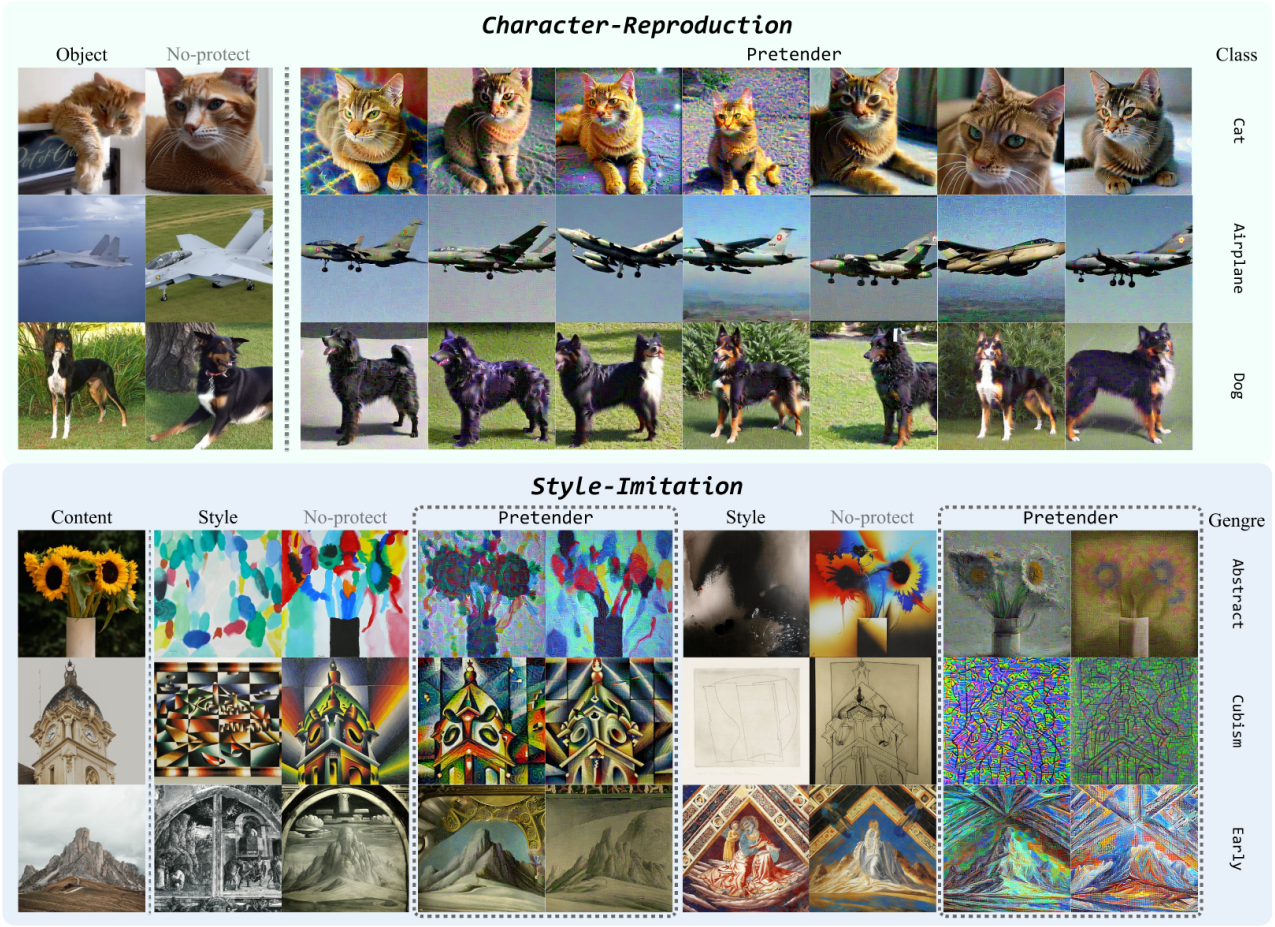

Figure 13: **Real-world performance evaluations on `scenario` [57].**

Takeaway 4. `Pretender` has demonstrated its effectiveness in real-world application scenarios.

11 Limitations and Discussions

Cross-Architecture Defense: The generalization evaluation in Sec. 8.3 highlights that `Pretender` still has room for improvement in defending against DMs with significant architectural variations. Considering the scalability of the `Pretender` framework, leveraging more advanced DMs (e.g., SDXL or those based on DiT) for optimizing protective noise, or adopting a mixed-expert training strategy, could be promising directions for enhancing cross-architecture defense capabilities.
Generalize to More Potential DFAs: `Pretender`'s defense induces shifts in the DM feature space, disrupting the **LR**. It has the potential to transfer to **SVDiff** [12], since SVDiff similarly directly adjusts the backbone UNet. `Pretender` is effective against **DB**, suggesting it could also interfere with the training of **TRL** [10] and **DraFT** [13] which employs additional loss functions to update the network. Lastly, `Pretender` demonstrates the strongest defense against **TI**, showing its ability to effectively disrupt word embeddings during training, making it likely to transfer to similar **Hifi-Tuner** [11].
Time Efficiency: `Pretender` incurs a slightly higher time cost compared to some approaches (e.g., `AdvDM`). We attribute this to the trade-off between time efficiency and protective performance. As demonstrated in our work, static surrogate model strategies struggle with generalization, and thus, updating surrogate models inevitably introduces additional time expenditures. In the future, we need to explore approaches to reduce the number of optimizaiton iterations for improving computational efficiency.

12 Conclusion

Our research addresses the critical challenge of safeguarding personal images from Diffusion Finetuning Attacks (DFAs) in the AI-creation era. We conceptualize active defense as a universal bi-level optimization problem, leading to the development of the `Pretender` defense algorithm. Comprehensive testing confirms `Pretender`'s success in preventing DFA-based imitation of unique visual patterns in personal image, proving its practical efficacy in real-world scenarios.

Acknowledgments

We sincerely thank the anonymous shepherd and all the reviewers for their constructive comments and suggestions. This work was supported in part by the National Key R&D Program of China under Grant 2023YFB2704700, in part by National Natural Science Foundation of China under Grant 62472276, in part by Shanghai Committee of Science and Technology, China (23511101000, 24BC3200400), and in part by Science and Technology Project of the State Grid Corporation of China under Grant 5700-202321603A-3-2-ZN.

References

[1] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

[2] Guinness Harry. The best ai image generators in 2023. zapier.com, May 2023. https://zapier.com/blog/best-ai-image-generator/.

[3] Midjourney Inc. Midjourney. https://www.midjourney.com/home/.

[4] stability.ai. Dreamstudio (stable diffusion). https://dreamstudio.ai/.

[5] civitai.com. Chilloutmix. https://civitai.com/models/6424/chilloutmix.

[6] Deck Andrew. Ai-generated art sparks furious backlash from japan's anime community. https://restofworld.org/, Oct. 2022. https://restofworld.org/2022/ai-backlash-anime-artists/.

[7] cloneofsimo. Low-rank adaptation for fast text-to-image diffusion fine-tuning. https://github.com/cloneofsimo/lora.

[8] 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.

[9] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22500–22510, 2023.

[10] Kevin Clark, Paul Vicol, Kevin Swersky, and David J Fleet. Directly fine-tuning diffusion models on differentiable rewards. *arXiv preprint arXiv:2309.17400*, 2023.

[11] Zhonghao Wang, Wei Wei, Yang Zhao, Zhisheng Xiao, Mark Hasegawa-Johnson, Humphrey Shi, and Tingbo Hou. Hifi tuner: High-fidelity subject-driven fine-tuning for diffusion models. *arXiv preprint arXiv:2312.00079*, 2023.

[12] Ligong Han, Yinxiao Li, Han Zhang, Peyman Milanfar, Dimitris Metaxas, and Feng Yang. Svdiff: Compact parameter space for diffusion fine-tuning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7323–7334, 2023.

[13] Sayak Paul, Kashif Rasul, and Leandro von Werra. Fine-tune stable diffusion models with ddpo via trl, 2023. https://huggingface.co/blog/trl-ddpo.

[14] Hugging Face. Diffusers. https://huggingface.co/docs/diffusers/index.

[15] AUTOMATIC1111. Stable diffusion web ui. https://github.com/AUTOMATIC1111/stable-diffusion-webui.

[16] 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. *USENIX Security Symposium*, 2023.

[17] 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*, 2023.

[18] Micah Goldblum, Dimitris Tsipras, Chulin Xie, Xinyun Chen, Avi Schwarzschild, Dawn Song, Aleksander Mądry, Bo Li, and Tom Goldstein. Dataset security for machine learning: Data poisoning, backdoor attacks, and defenses. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(2):1563–1580, 2022.

[19] Liam Fowl, Micah Goldblum, Ping-yeh Chiang, Jonas Geiping, Wojciech Czaja, and Tom Goldstein. Adversarial examples make strong poisons. *Advances in Neural Information Processing Systems*, 34:30339–30351, 2021.

[20] Pedro Sandoval-Segura, Vasu Singla, Jonas Geiping, Micah Goldblum, Tom Goldstein, and David Jacobs. Autoregressive perturbations for data poisoning. *Advances in Neural Information Processing Systems*, 35:27374–27386, 2022.

[21] Shawn Shan, Emily Wenger, Jiayun Zhang, Huiying Li, Haitao Zheng, and Ben Y Zhao. Fawkes: Protecting privacy against unauthorized deep learning models. In *USENIX Security Symposium*, 2020.

[22] Zheng Li, Ning Yu, Ahmed Salem, Michael Backes, Mario Fritz, and Yang Zhang. Unganable: Defending against gan-based face manipulation. *USENIX Security Symposium*, 2022.

[23] Stability AI. Stable diffusion public release. https://stability.ai/, Aug. 2022. https://stability.ai/blog/stable-diffusion-public-release.

[24] openai.com. Dall·e 2. https://openai.com/dall-e-2.

[25] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021.

[26] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022.

[27] Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting using denoising diffusion probabilistic models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11461–11471, 2022.

[28] Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. Sdedit: Guided image synthesis and editing with stochastic differential equations. In *International Conference on Learning Representations*, 2021.

[29] Lvmin Zhang and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. *arXiv preprint arXiv:2302.05543*, 2023.

[30] 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.

[31] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention*, pages 234–241. Springer, 2015.

[32] Sicong Han, Chenhao Lin, Chao Shen, Qian Wang, and Xiaohong Guan. Interpreting adversarial examples in deep learning: A review. *ACM Computing Surveys*, 2023.

[33] Hongshuo Liang, Erlu He, Yangyang Zhao, Zhe Jia, and Hao Li. Adversarial attack and defense: A survey. *Electronics*, 11(8):1283, 2022.

[34] Jie Zhang, Bo Li, Jianghe Xu, Shuang Wu, Shouhong Ding, Lei Zhang, and Chao Wu. Towards efficient data free black-box adversarial attack. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15115–15125, 2022.

[35] Ye Liu, Yaya Cheng, Lianli Gao, Xianglong Liu, Qilong Zhang, and Jingkuan Song. Practical evaluation of adversarial robustness via adaptive auto attack. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15105–15114, 2022.

[36] Shaopeng Fu, Fengxiang He, Yang Liu, Li Shen, and Dacheng Tao. Robust unlearnable examples: Protecting data against adversarial learning. In *International Conference on Learning Representations*, 2022.

[37] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems*, 35:36479–36494, 2022.

[38] Boris Dayma, Suraj Patil, Pedro Cuenca, Khalid Saifullah, Tanishq Abraham, Phuc Le Khac, Luke Melas, and Ritobrata Ghosh. Dall·e mini, 7 2021.

[39] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023.

[40] Enze Xie, Junsong Chen, Junyu Chen, Han Cai, Haotian Tang, Yujun Lin, Zhekai Zhang, Muyang Li, Ligeng Zhu, Yao Lu, et al. Sana: Efficient high-resolution image synthesis with linear diffusion transformers. *arXiv preprint arXiv:2410.10629*, 2024.

[41] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4195–4205, 2023.

[42] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *International Conference on Machine Learning*, 2024.

[43] Nilaksh Das, Madhuri Shanbhogue, Shang-Tse Chen, Fred Hohman, Siwei Li, Li Chen, Michael E Kounavis, and Duen Horng Chau. Shield: Fast, practical defense and vaccination for deep learning using jpeg compression. In *ACM SIGKDD International Conference on*

Knowledge Discovery & Data Mining, pages 196–204, 2018.

[44] Zhuoran Liu, Zhengyu Zhao, Alex Kolmus, Tijn Berns, Twan van Laarhoven, Tom Heskes, and Martha Larson. Going grayscale: The road to understanding and improving unlearnable examples, 2021.

[45] Aamir Mustafa, Salman H Khan, Munawar Hayat, Jianbing Shen, and Ling Shao. Image super-resolution as a defense against adversarial attacks. *IEEE Transactions on Image Processing*, 29:1711–1724, 2019.

[46] Chuan Guo, Mayank Rana, Moustapha Cisse, and Laurens Van Der Maaten. Countering adversarial images using input transformations. *arXiv preprint arXiv:1711.00117*, 2017.

[47] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

[48] Fisher Yu, Ari Seff, Yinda Zhang, Shuran Song, Thomas Funkhouser, and Jianxiong Xiao. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint arXiv:1506.03365*, 2015.

[49] Steven R Livingstone and Frank A Russo. The ryerson audio-visual database of emotional speech and song (ravdess): A dynamic, multimodal set of facial and vocal expressions in north american english. *PloS one*, 13(5):e0196391, 2018.

[50] kaggle.com. Painter by numbers, wikiart. https://www.kaggle.com/c/painter-by-numbers.

[51] Pexels. Open-source photos. https://www.pexels.com/.

[52] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in Neural Information Processing Systems*, 30, 2017.

[53] Tuomas Kynkäänniemi, Tero Karras, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Improved precision and recall metric for assessing generative models. *Advances in Neural Information Processing Systems*, 32, 2019.

[54] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.

[55] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International Conference on Machine Learning*, pages 12888–12900. PMLR, 2022.

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

[57] Scenario. Craft unique and style-consistent game assets with custom-trained ai models. https://www.scenario.com/.

[58] Guanzhong Chen, Zhenghan Qin, Mingxin Yang, Yajie Zhou, Tao Fan, Tianyu Du, and Zenglin Xu. Unveiling the vulnerability of private fine-tuning in split-based frameworks for large language models: A bidirectionally enhanced attack. In *ACM SIGSAC Conference on Computer and Communications Security*, pages 2904–2918, 2024.

Ethics

Our work is geared towards the protection of personal image data. The design of our approach does not involve intrusive attacks on privacy, and we have ensured that all data collected during the experiments come from reliable, authorized, and secure datasets. During human-based evaluations, the research team has carefully reviewed all the images presented to the participants to ensure that they do not contain harmful or inappropriate information. Therefore, we believe that our work adheres to ethical standards.

Open Science Policy

Our research artifacts are available at https://zenodo.org/records/14748741. The artifacts (.zip) include:

- **codes:** Core implementation of the proposed defense algorithm: `Pretender`.
- **datasets:** The dataset constructed in this work, including **LSUN**, **RAVDESS** and **WikiArt**.
- **protections:** The images protected by the `Pretender` algorithm correspond one-to-one with the groups in the dataset.

The codes along with detailed documentation and instructions will be open-sourced at https://github.com/fredericksz k/Pretender.

Appendix

A Evaluations

A.1 Semantic Similarity Metric

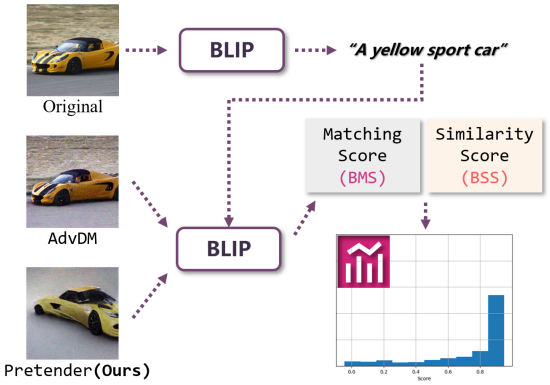

Figure 14: Illustration for calculating the BLIP-Matching-Score (**BMS**) and BLIP-Similarity-Score (**BSS**) utilizing the BLIP [55]

For the semantic similarity evaluations, due to the lack of universally applicable metrics in previous studies, we have designed two metrics named BLIP-Matching-Score (**BMS**) and BLIP-Similarity-Score (**BSS**). As illustrated in Fig. 14, we adopt BLIP [55] to caption the original image in a group, extract the most common descriptive texts, and use BLIP again to calculate the matching scores and cosine similarity score between the text and generated images. We further perform statistical analysis and report the mean/median of BMS, and mean of BSS.

A.2 Implementation Details

We primarily adhered to the initial settings recommended in the `diffuser` official documentation when employing the three algorithms for fine-tuning the Diffusion model. During the training process using **LR** algorithm, we designated the control generation prompt for **LSUN** dataset as *"a photo of sks ✯"* with '✯' being the current class name, such as "cat" "train" etc. For the **RAVDESS** dataset, we uniformly set the prompt as *"a photo of sks face"* As for the **WikiArt** dataset, we set the prompt as "a painting in the style of sks ✯" with '✯' referring to the current artistic genre name. We set the batch-size to 1 and the learning rate to 5e-4.

For the training process using **TI** algorithm, we set the special token in the control prompt as "<✯>" such as *"a photo of <car>"* or *"a painting in the style of <abstract>"*. We set the batch-size to 5 and the learning rate to 1e-4.

For the training process using **DB** algorithm, we adopted the same control generation prompt mode as LR. The prompt for auxiliary class-prior-preservation images was the control generation prompt with the word *"sks"* removed. We set the batch-size to 1 and the learning rate to 5e-4.

During the training process, all images were center-cropped and resized to 512×512. We used 100 epochs for the training steps. Additionally, we set the learning rate to 5e-5 for the RAVDESS dataset because we found that this dataset was more sensitive to hyper-parameters and large learning rates could result in overfitting of the trained model.

When implementing `Pretender`, we adopt LR to fine-tune the model parameters. The hyper-parameters, such as prompt, batch size, and learning rate settings for for each algorithm, remained the same as the LR training settings mentioned above. L_∞-Norm was used for applying adversarial perturbations, with a single step attack strength of $1/255$ and a perturbation budget of $8/255$. We set the number of protection optimization epochs to 150 and checked the model's loss every five epochs, saving the protected images with the highest losses.

As a baseline comparison, we also implemented and evaluated the `AdvDM` [17] algorithm. We reproduced the algorithm according to its original configurations reported in the paper. We adopted the same attack strength and attack budget as in `Pretender`. Besides we follow their 40 epochs attack iteration protocol.

For the evaluations of `Glaze` [16], we used their released executable program and adopted the officially recommended default settings (default intensity and medium render quality).

A.3 Perturbation Budget

Figure 15: **The effect of various purturbation budget.** We present the image with the lowest user rating to better illustrate the variations in noise.

The perturbation budget is the maximum intensity of noise added to an image. For example, if we use the l_∞ distance to restrict the perturbation and set the budget to be L. It means that the maximum value of each disturbed pixel in the image will not exceed L, and there is no limit to the number of pixels disturbed or the total amount of pixel disturbance value. The size of the perturbation budget clearly affects the

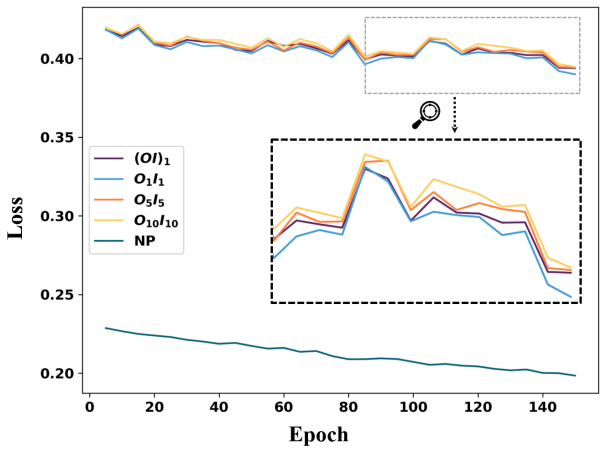

Figure 16: **The comparison between the proposed SGBP strategy($(OI)_1$) and several possible defending strategies.** NP refers to "No-Protection". The larger loss during the training indicates a better defense effect.

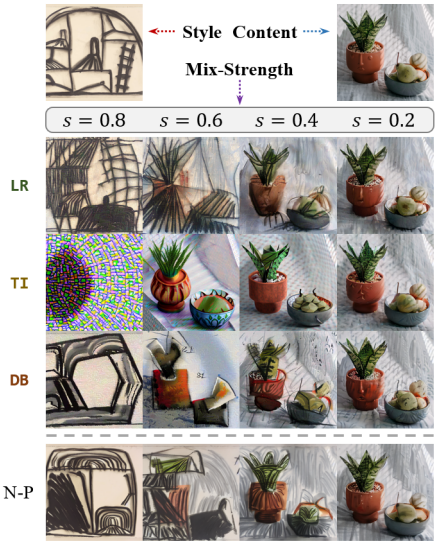

Figure 17: **Image protection effect on style-imitation under different mix-strength settings**. NP stands for "No-Protection". The comparative results demonstrate that our protection is effective across divers settings.

usability of the protected image. Therefore, in this section, we explore the performance of different perturbation budgets in our protection framework.

As shown in Fig. 15, the noise intensity in the image increases, becoming perceptible, and the loss during defense rises as the perturbation budget increases. Conversely, the model's final loss during training increases, and the quality of the generated image declines as the budget increases on the attacker's side. Notably, we observe a significant improvement in defense effectiveness at L within the range of $8/255$ to $16/255$. Therefore, defenders can make appropriate trade-offs.

B Method

B.1 Derivation of Bi-level Optimization

Firstly we focus on the outer-level optimization objective. For simplicity, we mark $x' = x + \delta$ and replace the $\theta(\delta)$ with θ because it's unchanged here:

$$\delta = \arg\min_{\delta} p_\theta(x'). \tag{12}$$

To minimize the $p_\theta(x')$, we decompose it into the integral form:

$$p_\theta(x') = \int p_\theta(x'_{0:T}) \mathrm{d}x'_{1:T}, \tag{13}$$

where $p_\theta(x'_{0:T})$ refers to the backward denoising process that reconstructs the fully-noised image x'_T to x'_0. x'_0 can be understood as a sample drawn from a hypothetical distribution A, which visually resembles x' to a great extent (implying similarity with the original image x as well). Our objective here is to disrupt the formation of this distribution, as described in Eq. (12).

Besides, we also note that the forward noising process DM is independent, i.e., the posterior $q(x'_{1:T}|x'_0)$ is fixed as a Gaussian distribution. Therefore, we could derive that:

$$\begin{aligned} \min \mathbb{E}\left[p_\theta(x')\right] &= \max \mathbb{E}\left[-\log p_\theta(x')\right] \\ &= \max \mathbb{E}\left[-\log p_\theta(x'_{0:T})\right] \\ &= \max \mathbb{E}\left[-\log \frac{p_\theta(x'_{0:T})}{q(x'_{1:T}|x'_0)}\right]. \end{aligned} \tag{14}$$

The fraction $-\log \frac{p_\theta(x'_{0:T})}{q(x'_{1:T}|x'_0)}$ is exactly the variational bound. According to the deduction of DM's training [1], we could finally obtain that:

$$\delta = \arg\max_{\delta} L(\theta(\delta), x+\delta). \tag{15}$$

And the inner optimization objective could be derived using the same approach.

B.2 Proof of Proposition 1

This section provides the proof of the Proposition 1 in Sec. 4.2.

Proof. Assume that the defense strategy starts with a static surrogate model $\theta(0)$. It calculates the protective noise through:

$$\delta^1 = \arg\max_{\delta} L(\theta(0), x+\delta). \tag{16}$$

At this point, the image released by the defender, x, together with its corresponding defense δ^1, is collected by the attacker

and subjected to a round of optimization using the inner-level optimization objective Eq. (7), resulting in network parameters $\theta_t(\delta^1)$ that satisfy:

$$\theta_t(\delta^1) = \arg\min_{\theta_t} L(\theta_t, x+\delta^1). \tag{17}$$

Ideally, the goal of bi-level optimization is to achieve $\max L(\theta_t(\delta^1), x+\delta^1)$. However, there is a clear contradiction with Eq. (17). Notably, $\theta_t(\delta^1)$ is typically trained by the attacker based on a pre-trained DM $\theta_t(0)$, and therefore may not converge to the pre-trained weights $\theta(0)$ in Eq. (16).

□

B.3 Analysis of Dynamic Method's Optimality

Proof. We need to prove that the following dynamic approch outperforms the static-surrogate strategy,

$$\theta(0) \to \delta^1 \to \theta(\delta^1) \to \delta^2 \to \theta(\delta^2) \to ... \to \delta^N, \tag{18}$$

i.e., we need to prove that δ^N is superior to δ^1. To simplify the proof, we demonstrate that δ^2 outperforms δ^1, from which subsequent conclusions can be deduced recursively. Meanwhile, we consider the optimistic scenario where the attacker's $\theta_t(0)$ equals the defender's surrogate model $\theta(0)$.

To define optimality, we consider that greater deviation induced in the attacker's model indicates better protection performance, i.e., proving:

$$L(\theta(0), x+\delta^2) > L(\theta(0), x+\delta^1) \tag{19}$$

Considering the small difference between δ^1 and δ^2, we use Taylor expansion for approximate analysis:

$$L(\theta(0), x+\delta^2) \approx L(\theta(0), x+\delta^1) + \nabla_\delta L(\theta(0), x+\delta^1)^\top (\delta^2 - \delta^1) \tag{20}$$

We further analyze δ^2. Since it is obtained by the defender through the model $\theta(\delta^1)$ trained on δ^1, it can be approximately expressed as:

$$\delta^2 = \delta^1 + \nabla_\delta L(\theta(\delta^1), x+\delta^1). \tag{21}$$

Here, we ignore the learning rate as they are all positive numbers. Substituting Eq. (20) and Eq. (21) into Eq. (19), the proof objective becomes:

$$\nabla_\delta L(\theta(0), x+\delta^1)^\top \nabla_\delta L(\theta(\delta^1), x+\delta^1) > 0. \tag{22}$$

The above equation indicates that the gradient of $\theta(0)$ with respect to the perturbation (i.e., the input image) and the gradient of $\theta(\delta^1)$ are positively correlated. This gradient direction simultaneously represents the network's vulnerability or defect pattern.

Since this is a fine-tuning process, the output pattern of the network parameters does not experience drastic changes, as explored in related works [58]. Moreover, although $\theta(\delta^1)$ undergoes some repairs compared to $\theta(0)$, these adjustments are localized and minor, targeting only a single sample, and are insufficient to address all defects, particularly in the early stages of training. Therefore, these two gradients could have a high degree of correlation, thereby proving the above equation.

□