Step Vulnerability Guided Mean Fluctuation Adversarial Attack against Conditional Diffusion Models

Hongwei Yu[1], **Jiansheng Chen**[1*], **Xinlong Ding**[1], **Yudong Zhang**[2], **Ting Tang**[1], **Huimin Ma**[1]
[1]School of Computer and Communication Engineering, University of Science and Technology Beijing, China
[2]Department of Electronic Engineering, Tsinghua University, China
yuhongwei22@xs.ustb.edu.cn, jschen@ustb.edu.cn, dingxl22@xs.ustb.edu.cn, zhangyd16@mails.tsinghua.edu.cn,
m202220901@xs.ustb.edu.cn, mhmpub@ustb.edu.cn

Abstract

The high-quality generation results of conditional diffusion models have brought about concerns regarding privacy and copyright issues. As a possible technique for preventing the abuse of diffusion models, the adversarial attack against diffusion models has attracted academic attention recently. In this work, utilizing the phenomenon that diffusion models are highly sensitive to the mean value of the input noise, we propose the Mean Fluctuation Attack (MFA) to introduce mean fluctuations by shifting the mean values of the estimated noises during the reverse process. In addition, we reveal that the vulnerability of different reverse steps against adversarial attacks actually varies significantly. By modeling the step vulnerability and using it as guidance to sample the target steps for generating adversarial examples, the effectiveness of adversarial attacks can be substantially enhanced. Extensive experiments show that our algorithm can steadily cause the mean shift of the predicted noises so as to disrupt the entire reverse generation process and degrade the generation results significantly. We also demonstrate that the step vulnerability is intrinsic to the reverse process by verifying its effectiveness in an attack method other than MFA. Code and Supplementary is available at https://github.com/yuhongwei22/MFA

Introduction

Due to the high generation quality and training stability, the diffusion model has become a competitive deep generation model recently (Ho, Jain, and Abbeel 2020; Song, Meng, and Ermon 2020; Croitoru et al. 2023; Rombach et al. 2022). A Diffusion model consists of two essential processes. The forward process is a Markov chain that gradually incorporates noises into the input data to diffuse it to a standard Gaussian noise. Conversely, the reverse process functions as a parametric Markov chain that runs in the opposite direction and is designed to learn how to reverse the diffusion process by estimating the added noises. To date, diffusion models have demonstrated outstanding performances by achieving many state-of-the-art results in various generation tasks.

To achieve better control over the generation during the reverse process, various prompts are used in diffusion models, such as images (Rombach et al. 2022; Batzolis et al. 2021; Gal et al. 2022), sketches (Voynov, Aberman, and Cohen-Or 2022; Peng et al. 2023), and text (Nichol et al. 2021; Poole et al. 2022; Ramesh et al. 2022; Saharia et al. 2022a). These prompts are encoded by a prompt encoder and serve as conditional inputs to each step of the reverse process, enabling effective control over the generation. Due to its stable theoretical foundation (Song et al. 2020; Bao et al. 2022) and highly applicable techniques (Gal et al. 2022; Lu et al. 2022), conditional diffusion models have been successfully used in diverse fields, including image synthesis (Ho, Jain, and Abbeel 2020; Song, Meng, and Ermon 2020; Song and Ermon 2019; Ruiz et al. 2023), image editing (Kawar et al. 2023; Batzolis et al. 2021; Esser et al. 2021), and video synthesis (Yang, Srivastava, and Mandt 2022). However, with the successful application of the conditional diffusion model, there has been a concern that its high-quality generation results may bring about privacy and copyright issues. Therefore, researchers (Salman et al. 2023; Liang et al. 2023; Zhuang, Zhang, and Liu 2023) are beginning to study the adversarial attack against diffusion models as a possible technique for preventing the abuse of diffusion models.

Existing research (Liang et al. 2023; Zhang et al. 2023) has revealed that conditional input is probably a weak point of the conditional diffusion model. This is because conditional diffusion models usually feed conditions to each step of the reverse process so that attackers can effectively influence the reverse process by adding adversarial perturbations to prompts. Most previous adversarial attacks against conditional diffusion models mainly focus on attacking the prompt encoder and the internal structure of Unet (Zhuang, Zhang, and Liu 2023; Zhang et al. 2023), known as the embedding attack. The core of such an attack is to increase the distance between the clean condition input and the corresponding adversarial example in the embedding space. As such, the embedding attack is more likely to attack the prompt encoder rather than the whole diffusion model since the reverse denoising process is basically not involved. Recently, there have been works (Liang et al. 2023; Liu et al. 2023) that consider the adversarial attack against the reverse process. A typical attacking strategy is to increase the error of the estimated noise in the reverse process. For example, AdvDM (Liang et al. 2023) increases the estimation error of the noise by directly maximizing the training loss of the diffusion model. Nevertheless, such an approach treats the

[*]Corresponding author

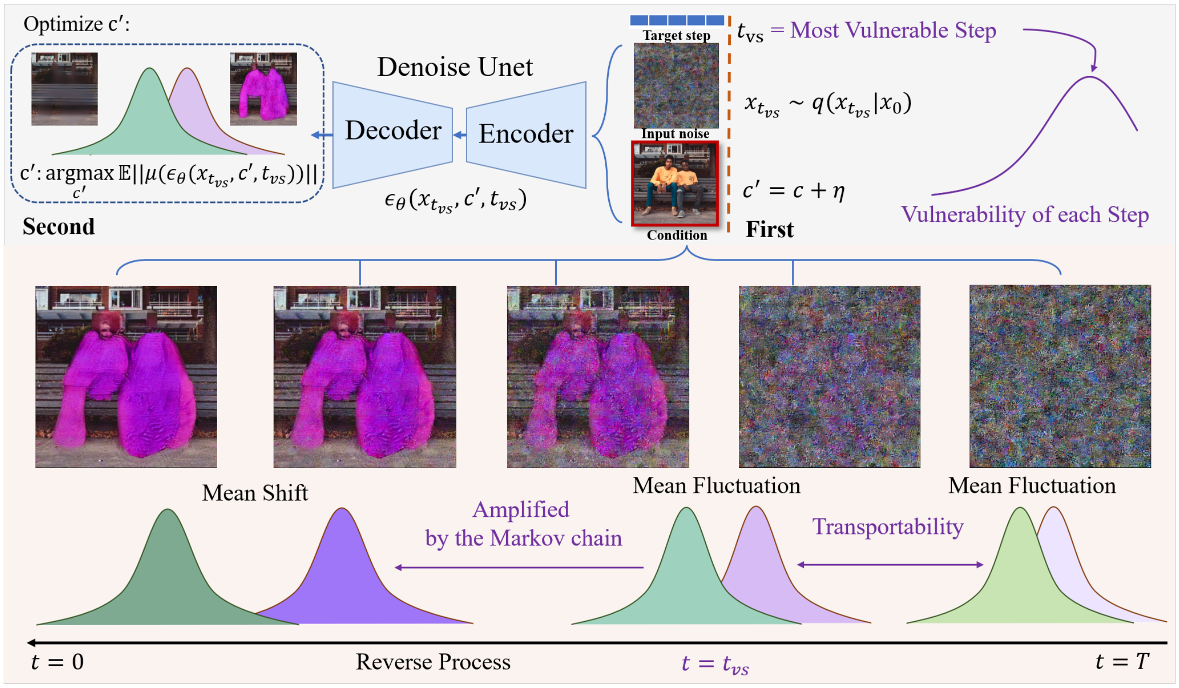
Optimize c′:
c′: $\underset{c'}{\text{argmax}}\,\mathbb{E}||\mu(\epsilon_\theta(x_{t_{vs}}, c', t_{vs}))||$
Second
Denoise Unet
Decoder
Encoder
$\epsilon_\theta(x_{t_{vs}}, c', t_{vs})$
Target step
Input noise
Condition
t_{vs} = Most Vulnerable Step
$x_{t_{vs}} \sim q(x_{t_{vs}}|x_0)$
$c' = c + \eta$
First
Vulnerability of each Step
Mean Shift
Amplified
by the Markov chain
Mean Fluctuation
Transportability
Mean Fluctuation
$t = 0$
Reverse Process
$t = t_{vs}$
$t = T$

Figure 1: Overview of MFA-MVS for generating adversarial examples against conditional diffusion models. The algorithm consists of two main parts. First, the adversarial vulnerabilities of different reverse steps are estimated. Second, the most vulnerable step is specified as the target for generating adversarial examples. The generated examples are sent to every step of the reverse process to generate fluctuation of the noise means.

reverse process as a black box which does not help to understand the differences and correlations between steps in the reverse process and limits the effectiveness of the attack.

In this paper, we focus on studying how conditional input adversarial samples influence the reverse process. We reveal that the reverse process steps are extremely sensitive to the mean value of the input noise. For example, if there is a 10% shift in the mean of the initial randomly sampled Gaussian noise input, the reverse process of a diffusion model will experience a collapse by generating a blank image without textures. Utilizing this phenomenon, we propose the Mean Fluctuation Attack (MFA) to introduce mean fluctuations during the reverse process. Adversarial examples generated by MFA can effectively influence the reverse process by shifting the mean values of the estimated noises.

Since the reverse process consists of multiple steps, it is usually necessary to specify a target step to attack in each iteration of the optimization process for generating the adversarial example against diffusion models. In existing works (Zhang et al. 2023; Xue et al. 2023; Liang et al. 2023), the target reverse steps are often uniformly randomly sampled. However, we argue that this may not be the best choice by revealing that the vulnerability against adversarial attacks of different reverse steps actually varies significantly. By appropriately modeling the adversarial vulnerability of reverse steps and increasing the probability of sampling the steps with higher vulnerability, the effectiveness of adversarial attacks can be substantially enhanced. Even more interestingly, we find that under certain conditions, the most effective adversarial attack can be achieved by attacking the most vulnerable reverse step only. We refer to such an attack method as MFA-MVS (Most Vulnerable Step), of which the algorithm flow is shown in Figure 1. Generally, the algorithm consists of two main parts. First, the adversarial vulnerabilities of different reverse steps are estimated. Second, the most vulnerable step is specified as the target for generating adversarial examples. We further propose a mathematical explanation for the vulnerability of different steps to reveal how adversarial samples generate mean value shifts of estimated noises for different steps under MFA attacks. We focus on attacking images as prompts in this work. However, our proposal can also be extended to other types of prompts in conditional generations using DMs. Extensive experiments are performed to verify that our proposal successfully steers diffusion models to generate mean shifts in the estimated noises, which ultimately degrades the generation quality to a significant extent. The main contributions of this work are as follows.

- We propose the Mean Fluctuation Attack (MFA) against conditional diffusion models based on the finding that the reverse process of the diffusion model is extremely sensitive to the shift of the mean noise value.
- We reveal that reverse steps differ a lot in terms of the vulnerability against adversarial attacks, based on which the effectiveness of MFA can be further enhanced.
- We provide a mathematical explanation on the adversarial vulnerability of the reverse steps against MFA.

Preliminary

Diffusion Models

The Diffusion Model (DM) is a latent variable model of the form $p_\theta(x_0) := \int p_\theta(x_0 : T) dx_{1:T}$, where $x_1, ..., x_T$ are latent variables of the same dimensionality as the data $x_0 \sim q(x_0)$. The joint distribution $p_\theta(x_{0:T})$ is called the reverse process or the generation process, and it is defined as a Markov chain with learned Gaussian transitions starting at $p(x_T) = \mathcal{N}(x_T; 0, I)$. A DM usually contains two processes, namely the forward process and the reverse process.

Forward Process: What distinguishes diffusion models from other types of latent variable models is that the approximate posterior $q(x_{1:T}|x_0)$, called the forward process or diffusion process, is fixed to a Markov chain that gradually adds Gaussian noise to the data according to a variance schedule $\alpha_1, ..., \alpha_T$ as Eq. 1 , where $\alpha_t, \beta_t > 0$, $\alpha_t^2 + \beta_t^2 = 1$, $\bar{\alpha_t} = \alpha_t \alpha_{t-1} ... \alpha_1$, and $\bar{\beta}_t = \sqrt{1 - \bar{\alpha_t}^2}$.

$$\begin{aligned} q(x_{1:T}) &:= \prod_{t=1}^{T} q(x_t|x_{t-1}), \\ q(x_t|x_{t-1}) &= \mathcal{N}(x_t; \alpha_{t-1} x_{t-1}, \beta_{t-1}^2 I), \\ q(x_t|x_0) &= \mathcal{N}(x_t; \bar{\alpha}_t x_0, \bar{\beta}_t^2 I) \end{aligned} \tag{1}$$

Reverse Process: The $p(x_{t-1}|x_t, x_0)$ is usually used to approximate $p(x_{t-1}|x_t)$, which can be expressed as Eq. 2.

$$\begin{aligned} p_\theta(x_{0:T}) &:= p(x_T) \prod_{t=1}^{T} p_\theta(x_{t-1}|x_t), \\ p(x_{t-1}|x_t) &\approx p(x_{t-1}|x_t, x_0), \\ p(x_{t-1}|x_t, x_0) &= \mathcal{N}(x_{t-1}; \frac{1}{\alpha_t}(x_t - \frac{\beta_t^2}{\bar{\beta}_t} \epsilon_\theta(x_t, t)), \frac{\bar{\beta}_{t-1}^2 \beta_t^2}{\bar{\beta}_t^2} I) \end{aligned} \tag{2}$$

Conditional diffusion models add the condition c to each step to control the reverse process. The training loss function is shown in Eq. 3, and the aim is to predict the added noise $\varepsilon \sim \mathcal{N}(0, I)$ using the model $\epsilon_\theta(x_t, t, c)$.

$$L(\theta) := \mathbb{E}_{t, x_0, \epsilon, c} ||\varepsilon - \epsilon_\theta(x_t, t, c)||^2 \tag{3}$$

Adversarial Examples

Adversarial examples for classifiers: Given an image x and a classifier $f(\cdot)$ (Madry et al. 2017; Carlini and Wagner 2017), an adversarial example x' satisfies two properties: $\mathcal{D}(x, x')$ is small for some distance metric $\mathcal{D}$, and $f(x) \neq f(x')$. That is, images x and x' appear visually similar but x' is classified incorrectly. Following previous work (Dong et al. 2018; Zhang et al. 2022; Goodfellow, Shlens, and Szegedy 2014; Yu et al. 2023b), we use l_∞ as a distance matrix to measure the similarity between two images.

Adversarial examples for Diffusion models: Recently, many studies (Nie et al. 2022; Sun et al. 2022; Lee and Kim 2023) have employed Diffusion models as a supplementary technique to enhance the robustness of classification models. Nevertheless, there has been limited research conducted on the robustness of diffusion models or on adversarial attacks against conditional diffusion models. Early works mainly focus on attacking the prompt encoder (Maus et al. 2023; Millière 2022; Daras and Dimakis 2022). Zhang et al. (Zhuang, Zhang, and Liu 2023) manipulate the text encoder by including redundant characters in the input prompt to deceive it into attacking diffusion models. Zhang et al. (Zhang et al. 2023) aim to attack the the internal structure of Unet and distort the resulting image to disrupt the function of latent diffusion models. However, none of these studies delve into the adversarial robustness of the reverse denoising process, which is essential to DMs. Recently, there have been works that start to consider the adversarial attack against the reverse process, typically by increasing the error of the estimated noise. For example, AdvDM (Liang et al. 2023) attacks the reverse process by directly maximizing the training loss shown in Eq. 3, which represents the estimation error between the predicted noise and the added noise. This work mainly focuses on the textual inversion task in which several concept images provided by the user are used to learn pseudo-words in the space of text embedding to represent these concepts. Then these pseudo-words are combined into natural language sentences to guide the personalized generation. It is actually not clear whether AdvDM effectively attacks the word generation model's initial stage or the subsequent phase of the diffusion model. The differences and correlations between reverse steps are not explicitly studied in AdvDM which limits the effectiveness of the attack.

Methodology

In this section, we first introduce the Mean Fluctuation Attack (MFA). Then by analyzing the unique properties of DM, we model the vulnerability of different steps in the reverse process against adversarial attacks. We further demonstrate that more effective attack can be achieved by considering step vulnerability in MFA.

Mean Fluctuation Attack

We have observed that DMs are highly sensitive to mean values of the initial randomly sampled Gaussian noise input X_T. Based on such an observation, we propose the Mean Fluctuation Attack (MFA) that aims at generating mean fluctuations by increasing (or decreasing) the mean value of predicted noise in the reverse process. Suppose the clean prompt image used as the condition to be c, we define the adversarial example as $c' = c + \delta$, where δ is the adversarial perturbation. The objective of MFA is to maximize the mean fluctuation generated during the reverse process, which can be expressed as Eq. 4. More specifically, we aim to find the optimal perturbation δ that maximizes the expectation of the mean value of the predicted noise $\epsilon_\theta(x_t, t, c + \delta)$. Here, x_t is sampled from the distribution $q(x_t|x_0)$, t is sampled from the uniform distribution $U(1, T)$, and η is the norm constraint of perturbation δ. Detail flowchart of the algorithm is shown in the Supplementary.

$$\begin{aligned} &\delta := \arg\max_{\delta} \mathbb{E} ||\mu(\epsilon_\theta(x_t, t, c + \delta))||, \\ &where\ x_t \sim q(x_t|x_0), t \sim U(1, T), ||\delta||_\infty \leq \eta \end{aligned} \tag{4}$$

MFA can effectively generate mean fluctuations, which affect the reverse process and result in mean shifting phenomena similar to that of direct modification of x_T. Intuitively, the MFA adversarial example will result in a mean shift of the predicted noise with consistent direction for each step in the reverse process. The accumulation of such mean shifts can ultimately invalidate the generation results.

In the basic version of MFA, the reverse steps t are sampled with equal probability, assuming that the effectiveness for attacking each step is the same. However, we have discovered that when optimizing adversarial examples, the benefits of attacking different steps vary a lot. To further enhance the effectiveness of MFA, we combine theoretical analysis with empirical observations and introduce the concept of step-wise vulnerability, denoted as $v(t)$, which quantifies the benefit obtained by attacking step t. As such, we can increase the sampling probabilities of steps with higher vulnerability. Therefore, the objective of MFA can be modified as Eq. 5. We name such a modified version as MFA-VT.

$$\begin{aligned} &\delta := \arg\max_{\delta} \mathbb{E}||\mu(\epsilon_\theta(x_t, t, c+\delta))||, \\ &where\, x_t \sim q(x_t|x_0), t \sim P(t), ||\delta||_\infty \leq \eta \end{aligned} \tag{5}$$

In Eq. 5, $P(t)$ represents the sampling distribution determined by vulnerabilities. The probability of sampling step t is defined as Eq. 6.

$$P(t) = \frac{v(t)}{\sum_{i=1}^{T} v(i)} \tag{6}$$

We have further observed that when the number of attacking iterations is sufficiently large, MFA-VT produces excellent results. However, when the number of attacking iterations is limited, fixing t to the step with the highest vulnerability can effectively increase the performance, as is shown in Eq. 7. We name such a modified version as MFA-MVS.

$$\begin{aligned} &\delta := \arg\max_{\delta} \mathbb{E}||\mu(\epsilon_\theta(x_t, t, c+\delta))||, \\ &where\, x_t \sim q(x_t|x_0), t = \arg\max_{t} v(t), ||\delta||_\infty \leq \eta \end{aligned} \tag{7}$$

In the next subsection, we will present a detailed analysis as well as a mathematical modeling of $v(t)$.

Step Vulnerability

We believe that there are three main factors that contribute to the varying benefits of attacking different steps t in the reverse process. **(1) Chain-like structure.** In the reverse process, the input to a step depends on the output of the previous step. Hence, the mean fluctuations will be transferred and amplified step by step. **(2) Stability of different steps.** It can be observed that attacking different steps result in different magnitudes of mean fluctuations. Such a variability in stability also contributes to the varying benefits of attacking different steps. **(3) Transferability between different steps.** When an adversarial example generated by attacking a specific step is applied to other steps as condition , their effectiveness tends to diminish as the steps become more distant from the attacked step. The diminishing transferability between steps further adds to the variation in benefits obtained from attacking different steps.

Coupled together, these three factors contribute to the significant difference in the vulnerability of reverse steps against adversarial attacks. In the following, we first analyze the three factors separately by fixing the target step t to attack. Then we combine the three factors to present a unified mathematical model of $v(t)$. All the analyses in this section are conducted based on the inpainting task using the Latent diffusion model (LDM) (Rombach et al. 2022).

(1) Chain-like structure. A typical forward process of a denoising DM is a stable Markov chain that gradually adds Gaussian noise to the data based on pre-designed noise until the distribution of the data converges to the standard Gaussian distribution. According to Eq. 1, we can model the forward process as Eq. 8. Repeated iterations lead to the formula for diffusion process sampling as Eq. 9.

$$x_t = \alpha_t x_{t-1} + \beta_t \varepsilon_t, \varepsilon_t \sim N(0, I) \tag{8}$$

$$x_t = \bar{\alpha}_t x_0 + \bar{\beta}_t \bar{\varepsilon}_t, \bar{\varepsilon}_t \sim \mathcal{N}(0, I) \tag{9}$$

For an x_0, assume that the mean fluctuation occurs at the n_{th} step in the reverse process, denoted as $x'_n = x_n + \xi$, $1 \leq n \leq T$ and ξ is a constant. From Eq. 9, the correct x_0 should be $x_0 = \frac{1}{\bar{\alpha}_n}(x_n - \bar{\beta}_n \bar{\varepsilon}_n)$, and after the mean fluctuation occurs at the n_{th} step, the incorrectly estimated x'_0 can be expressed as $x'_0 = \frac{1}{\bar{\alpha}_n}(x_n + \xi - \bar{\beta}_n \bar{\varepsilon}_n)$. As such, the impact of the mean fluctuation occurs at the n_{th} step on the final generation result can be expressed as Eq. 10.

$$x'_0 - x_0 = \frac{\xi}{\bar{\alpha}_n} \tag{10}$$

Since $\bar{\alpha}_n = \prod_{i=1}^{n} \alpha_i, 0 < \alpha_i < 1$, it is obvious that mean fluctuations are amplified as the reverse process unfolds, and as n increases, this effect becomes more significant.

(2) Stability of different steps. If attacking different steps produce the fluctuations of the same magnitude, it is clear that attacking steps at larger t will yield greater benefits according to Eq. 10. However, we find that this is not true in practice. One reason is that in actual attacks, the magnitudes of mean fluctuation generated by attacking different steps differs. The conditional diffusion models estimates the added noise $\varepsilon \sim N(0, I)$ using $\epsilon(z_t, t, c)$, which can be rephrased as $\epsilon(\bar{\alpha}_t z_0 + \bar{\beta}_t \bar{\varepsilon}_t, t, c)$. It can be observe that as t increases, the first term of the input approaches noise, making it easier for the network to estimate the noise. Therefore, steps at larger t are substantially more stable when facing attacks. To confirm this, we calculated the training loss which measures the different between estimated noise and added noise at different steps and found that the loss decreases as t increases. Due to the complexity of neural networks, it is difficult to theoretically derive and model the stability of different steps. Therefore, we performed an empirical modeling by calculating the mean difference in predicted noise before and after the attack for different steps and normalized the results. Detail results are shown in the Supplementary, which reveals an approximately linear relationship between the stability and t. As t increases, the shift of the noise mean decreases, indicating stronger network stability. The stability can be approximated by a simple linear relationship defined in Eq. 11, of which the Goodness of Fit $R^2 \approx 0.972$, indicating a high degree of fit between $S(t)$ and the actual data.

$$S(t) = 1 - 0.8 * \frac{t}{T} \tag{11}$$

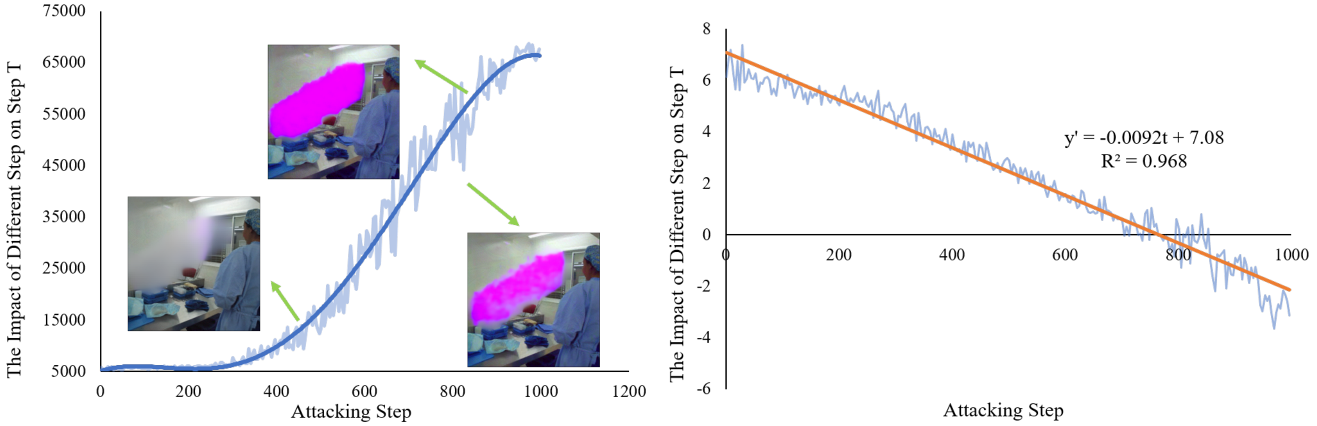

Figure 2: Statistical results on the impact of adversarial examples generated by attacking different steps on step T. The left figure shows that the attack becomes less effective when the attack step is farther from T. For the data in the left figure, we normalized it and fitted it linearly.

(3) Transferability between different steps. When attacking the condition of a DM, the generated adversarial example are actually passed to all the steps of the reverse process. Considering that different steps are using the same network with different inputs for noise prediction, the transferability of the adversarial example should also be considered. Specifically, an adversarial example generated by attacking a fixed target step should also have certain attacking effect on other steps. We further find that such transferability is more prominent between adjacent steps. This is understandable considering that the inputs of adjacent steps are highly similar. For example, since x_n are highly similar to x_{n-1} and x_{n+1}, the adversarial example c_n' generated through attacking the step $t = n$ will also have significant attack effects on steps $t = n-1$ and $t = n+1$. Intuitively, the further apart two steps are, the weaker the transferability will be.

To verify the conjecture, we applied the adversarial example c_t' generated by attacking step t and the clean sample c to step T. We calculate the difference $y = ||x_0' - x_0||_1$ between x_0' and x_0 predicted using c_t' and c. To better fit the data, we first normalized the data using a nonlinear transformations $y' = log((1/(y + e^{-5})) - 1)$. Subsequently, we visualized the relationship between y' and t and fitted it using a straight line. From the left of Figure 2, the experimental results verify our conjecture that the generated adversarial examples are more effective when being applied to steps closer to the target step. The relationship between y' and t can be represented as: $y' = -0.0092t + 7.08$. Therefore, the transferability of the adversarial examples generated by attacking step t applying on step i can be estimated as Eq. 12.

$$\begin{aligned}\tau_t^i &= \frac{1}{1 + e^{(7.08 - \frac{9.2(T - abs(t-i))}{T})}} \\ &= \frac{1}{1 + e^{(-2.08 + \frac{9.2abs(t-i)}{T})}}\end{aligned} \tag{12}$$

From the visualization results in Figure 2 left, we find that as the target step gets closer to T, the adversarial examples generated by attacking target step have better attack effects on step T. Figure 2 right shows the relationship between y' and t. We calculated the correlation coefficient between the fitted curve and the data points, obtaining a $R^2 \approx 0.968$, indicating a high degree of fit.

Step Vulnerability. Taking into account the three above factors, along with the coefficients in Eq. 2, we define the vulnerability of step t as the total magnitude of fluctuations that can be generated by attacking step t as Eq. 13, in which δ_t^i signifies the effect of adversarial examples generated at t step on the i step, $1/\bar{\alpha}_i$ represents the amplification effect brought by the chain-like structure, $\beta_i^2/\bar{\beta}_i$ is the coefficient in front of the reverse process shown in Eq. 2, and the last two terms represent transferability and stability respectively.

$$\begin{aligned}v(t) &= \sum_{i=1}^{T} \frac{1}{\bar{\alpha}_i} \frac{\beta_i^2}{\bar{\beta}_i} * \tau_t^i * S(t)\xi \\ &= \sum_{i=1}^{T} \frac{1}{\bar{\alpha}_i} \frac{\beta_i^2}{\bar{\beta}_i} * \frac{1}{1 + e^{(-2.08 + \frac{9.2abs(t-i)}{T})}} * (1 - \frac{0.8t}{T})\xi\end{aligned} \tag{13}$$

The fitting curve in the Supplementary verifies our hypothesis that there exists a trade-off when selecting the target step. We conducted experiments on multiple models and tasks. The results in Figure 4 aligns well with Eq. 13 that the highest vulnerability is typically observed at the latter stage of the reverse process, e.g. at step $t \approx 0.8T$ for LDM.

Experiments

Dataset and Experimental Settings

In this section, we evaluate our methods on multiple tasks. For the inpainting task, we utiliz the Places dataset (Zhou et al. 2017). For the super-resolution task, we employ the ImageNet dataset (Deng et al. 2009). Following existing research (Yu et al. 2023a; Shang et al. 2023) in adversarial examples, we use l_∞ norm as the constraint for generating the adversarial examples. We set per-step perturbation budget as 1/255, the total budget as 8/255, and attacking iterations as

	Random Mask Thick				Random Mask Medium				Random Mask Thin			
Method	FID↑	Delta E↑	PSNR↓	SSIM↓	FID↑	Delta E↑	PSNR↓	SSIM↓	FID↑	Delta E↑	PSNR↓	SSIM↓
No Attack	10.6	6.01	22.0	0.84	15.2	5.15	23.8	0.87	13.1	3.40	19.8	0.63
Embedding Attack	17.1	6.65	21.3	0.76	20.8	5.91	23.1	0.78	19.0	4.30	19.3	0.55
AdvDM	18.4	7.46	14.3	0.42	20.2	6.19	22.4	0.79	15.6	4.26	19.1	0.56
AdvDM-MVS	22.9	8.31	13.1	0.39	22.8	7.39	21.0	0.77	19.8	4.44	18.8	0.55
MFA	33.4	10.07	12.9	0.41	32.2	8.37	17.1	0.61	26.3	7.51	16.8	0.49
MFA-MVS	**52.5**	**12.77**	**11.8**	**0.40**	**46.8**	**10.79**	**14.8**	**0.59**	**29.1**	**8.70**	**15.9**	**0.47**
MFA-VT	44.9	11.89	12.1	0.41	41.8	9.96	15.5	0.60	28.2	8.64	16.4	0.48

Table 1: The attacking performance against conditional diffusion models on the inpainting task. The best attacking performances are marked as bold, while the second-best results are marked underline.

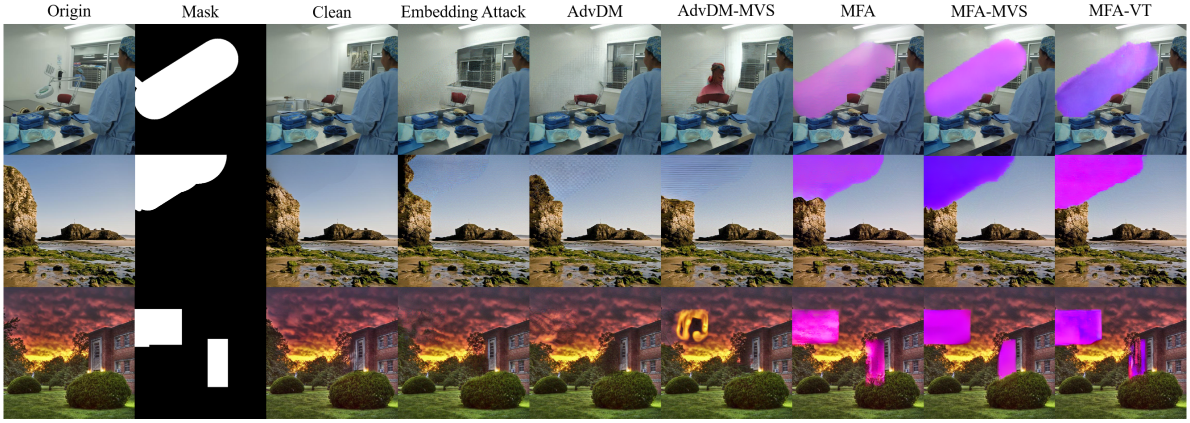

Figure 3: Inpainting results obtained by using adversarial examples generated by different attacks as condition inputs.

70. We conduct our experiments on inpainting and super-resolution tasks. We use 8 NVIDIA RTX 3090 GPUs for all experiments. More visualizations of experimental results are shown in the Supplementary. Superparametric experiments on the size of the total budget are also in the Supplementary.

Evaluation on Inpainting Task

We first evaluate the performance of the MFA algorithm on the inpainting task. Following the setup of the Latent Diffusion Model (LDM), the condition in the inpainting task consists of a mask m and an image x. To evaluate MFA quantitatively, we random select 2,000 images from Places365 (Zhou et al. 2017). The dataset preprocessing is the same as LaMa (Suvorov et al. 2022) and the detail of mask generation are shown in the Supplementary. By generating masks of different sizes on the dataset, we categorized them into three types, thick, medium, and thin. The implementation details for MFA on LDM are shown in the Supplementary.

We evaluate the inpainting quality by four metrics. Fréchet Inception Distance (FID) is a metric for quantifying the realism and diversity of images. Delta E (Sharma and Bala 2017) is a calculation of the change in color as measured in the Hunter Lab color space. Peak Signal-to-Noise Ratio (PSNR) is a widely used metric that measures the quality of a processed image by comparing it to the original. The structural similarity index measure (SSIM) is used as a metric to measure the similarity between given images.

Table 1 presents the quantitative results of our method on the inpainting task. From Table 1, it can be observed that our method can effectively attack conditional diffusion models. Also, attacking the most vulnerable step can not only significantly improve the performance of MFA, but also AdvDM. This indicates that as a guide, the step vulnerability can effectively enhance the performance of different attack methods. This verifies that step vulnerability is an intrinsic property of the reverse process.

From Figure 3, it can be observed that the images generated from clean samples are very similar to the surrounding scenery with no obvious differences. AdvDM can produce textures, with slight differences from the surroundings but no significant color variations. MFA can effectively influence the generation resulting in producing purple color for the inpainted regions, which is similar to directly modifying the mean of x_T. We will show the comparison between MFA and directly modifying x_T in the Supplementary material. Moreover, MFA-MVS further enhances the attack effect and effectively induces mean shift, generating anomalous color blocks different from other areas. MFA-VT also enhances the attack effect. However, when the number of attack iterations is small, the effect of MFA-VT is not as good as MFA-MVS. In the Section 4.4, it can be observed that when the number of iterations is sufficient, MFA-VT performs better than MFA-MVS, which also demonstrates the rationality and effectiveness of step vulnerabilities. We also evaluated our generated adversarial examples in the Supplementary using a basic JPEG compression defense method.

Method	PSNR↓	SSIM↓
No Attack	20.4	0.56
Embedding Attack	20.2	0.54
AdvDM	19.5	0.49
AdvDM-MVS	18.9	0.46
MFA	17.4	0.44
MFA-MVS	**16.3**	**0.37**
MFA-VT	16.9	0.41

Table 2: The attacking performance against conditional diffusion models on the super-resolution task. The best attacking performances of methods are marked as bold, while the second-best results are marked underline.

Evaluation on Super-resolution Task

We also validate our performance on the super-resolution task. Following the settings of LDM (Rombach et al. 2022), we evaluate the performance on ImageNet (Deng et al. 2009). The dataset preprocessing is same with SR3 (Saharia et al. 2022b). We randomly select 1,000 images to attack. The condition of super-resolution task is a low resolution image. We evaluate the super-resolution results using PSNR and SSIM, which are mentioned in section 4.2.

Table 2 presents the quantitative results of our method on the super-resolution task. The best result is highlighted in bold, and the second-best result is underlined. From the evaluation metrics, we can observe that our attack effectively reduces the generation quality of images, successfully attacking DMs in the super-resolution task, and also demonstrating the benefits of using the step vulnerability as guidance. The visualization results in the Supplementary show that our method has a more effective influence on the reverse process, resulting in more obvious noise and unreasonable textures in the super-resolution results.

Validation of Step Vulnerability

To further verify the effectiveness of step vulnerability, we conduct attacks on multiple models. We select three models, namely LDM-Inpainting, LDM-SR, and SR3, which all use images as conditions. To validate the effectiveness and universality of step vulnerability, we chose models that have different total number of reverse steps T. For LDM-Inpainting and LDM-SR we set T to 1000, while for SR3 we set T to 2000. We launch attacks on different steps of each model, generating adversarial samples for each step. For each model, we use 1000 images to calculate the PSNR metric which is then normalized negatively considering that the worse PSNR represents the better attack performance.

Figure 4 shows the normalized data collected from three different diffusion models represented in different colors. It can be observed that all models achieve maximum attack effectiveness at around $t = 0.8T$. This indicates that there is a strong consistency in terms of step vulnerability across different diffusion models. Moreover, the theoretic curve of the step vulnerability in Eq. 13 is shown as the solid line in Figure 4, indicating that our theoretical analysis is highly consistent with the actual situation.

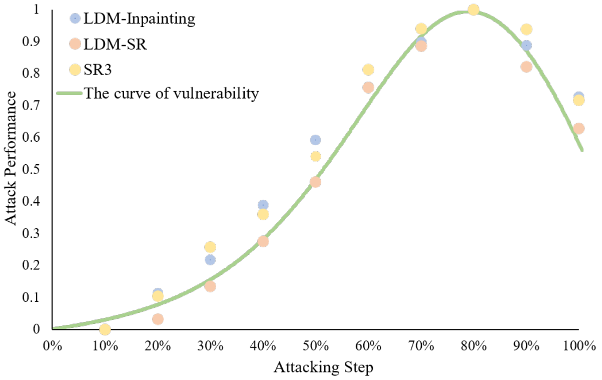

Figure 4: The statistical results on effects of attacking different steps on different models and tasks.

Attacking Iterations

The number of attacking iterations determines whether the generated adversarial examples can fit well with the step vulnerability curve we modeled, thus having a significant impact on the adversarial examples generated by MFA-VT. To investigate the impact of this hyperparameter, we conducte experiments on the Places365 dataset, following the experimental settings decribed in Section 4.1. We calculated the metrics for both MFA-MVS and MFA-VT under different attack step values ranging from 10 to 1000.

The results, as shown in Table 3, indicate that as the number of attack iterations increases, both MFA-MVS and MFA-VT show improved performance. However, with further increases in attacking iterations, MFA-VT surpasses MFA-MVS in terms of effectiveness. This suggests that choosing MFA-MVS has an advantage when the step value is small. Moreover, the experimental results further validate the effectiveness of the step vulnerability curve we modeled.

	MFA-VT		**MFA-MVS**	
Attacking iterations	FID↑	Delta E↑	FID↑	Delta E↑
10	17.9	7.58	19.1	8.06
70	44.9	11.89	52.5	12.77
1000	63.4	17.43	59.2	16.82

Table 3: Ablation on the number of attacking iterations

Conclusions

In this paper, we propose the Mean Fluctuation Attack (MFA) against conditional diffusion models based on the finding that the reverse process of the diffusion model is extremely sensitive to the shift of the mean noise value. We present that the attacking performance can be further enhanced under the guidance of the step vulnerability. We provide a mathematical explanation of the adversarial vulnerability of the reverse step against MFA. The experiments demonstrate that MFA can effectively influence the reverse process and choosing vulnerable steps to attack can further improve the attacking performance.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (62376024) and by the National Key R&D Program of China (2022ZD0117902).

References

Bao, F.; Li, C.; Zhu, J.; and Zhang, B. 2022. Analytic-dpm: an analytic estimate of the optimal reverse variance in diffusion probabilistic models. *arXiv preprint arXiv:2201.06503*.

Batzolis, G.; Stanczuk, J.; Schönlieb, C.-B.; and Etmann, C. 2021. Conditional image generation with score-based diffusion models. *arXiv preprint arXiv:2111.13606*.

Carlini, N.; and Wagner, D. 2017. Towards evaluating the robustness of neural networks. In *2017 ieee symposium on security and privacy (sp)*, 39–57. Ieee.

Croitoru, F.-A.; Hondru, V.; Ionescu, R. T.; and Shah, M. 2023. Diffusion models in vision: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Daras, G.; and Dimakis, A. G. 2022. Discovering the hidden vocabulary of dalle-2. *arXiv preprint arXiv:2206.00169*.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, 248–255. Ieee.

Dong, Y.; Liao, F.; Pang, T.; Su, H.; Zhu, J.; Hu, X.; and Li, J. 2018. Boosting adversarial attacks with momentum. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 9185–9193.

Esser, P.; Rombach, R.; Blattmann, A.; and Ommer, B. 2021. Imagebart: Bidirectional context with multinomial diffusion for autoregressive image synthesis. *Advances in neural information processing systems*, 34: 3518–3532.

Gal, R.; Alaluf, Y.; Atzmon, Y.; Patashnik, O.; Bermano, A. H.; Chechik, G.; and Cohen-Or, D. 2022. An image is worth one word: Personalizing text-to-image generation using textual inversion. *arXiv preprint arXiv:2208.01618*.

Goodfellow, I. J.; Shlens, J.; and Szegedy, C. 2014. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*.

Ho, J.; Jain, A.; and Abbeel, P. 2020. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33: 6840–6851.

Kawar, B.; Zada, S.; Lang, O.; Tov, O.; Chang, H.; Dekel, T.; Mosseri, I.; and Irani, M. 2023. Imagic: Text-based real image editing with diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6007–6017.

Lee, M.; and Kim, D. 2023. Robust evaluation of diffusion-based adversarial purification. *arXiv preprint arXiv:2303.09051*.

Liang, C.; Wu, X.; Hua, Y.; Zhang, J.; Xue, Y.; Song, T.; Zhengui, X.; Ma, R.; and Guan, H. 2023. Adversarial Example Does Good: Preventing Painting Imitation from Diffusion Models via Adversarial Examples.

Liu, Q.; Kortylewski, A.; Bai, Y.; Bai, S.; and Yuille, A. 2023. Intriguing Properties of Text-guided Diffusion Models. *arXiv preprint arXiv:2306.00974*.

Lu, C.; Zhou, Y.; Bao, F.; Chen, J.; Li, C.; and Zhu, J. 2022. Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps. *Advances in Neural Information Processing Systems*, 35: 5775–5787.

Madry, A.; Makelov, A.; Schmidt, L.; Tsipras, D.; and Vladu, A. 2017. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*.

Maus, N.; Chao, P.; Wong, E.; and Gardner, J. 2023. Adversarial prompting for black box foundation models. *arXiv preprint arXiv:2302.04237*.

Millière, R. 2022. Adversarial attacks on image generation with made-up words. *arXiv preprint arXiv:2208.04135*.

Nichol, A.; Dhariwal, P.; Ramesh, A.; Shyam, P.; Mishkin, P.; McGrew, B.; Sutskever, I.; and Chen, M. 2021. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*.

Nie, W.; Guo, B.; Huang, Y.; Xiao, C.; Vahdat, A.; and Anandkumar, A. 2022. Diffusion models for adversarial purification. *arXiv preprint arXiv:2205.07460*.

Peng, Y.; Zhao, C.; Xie, H.; Fukusato, T.; and Miyata, K. 2023. DiffFaceSketch: High-Fidelity Face Image Synthesis with Sketch-Guided Latent Diffusion Model. *arXiv preprint arXiv:2302.06908*.

Poole, B.; Jain, A.; Barron, J. T.; and Mildenhall, B. 2022. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988*.

Ramesh, A.; Dhariwal, P.; Nichol, A.; Chu, C.; and Chen, M. 2022. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*.

Rombach, R.; Blattmann, A.; Lorenz, D.; Esser, P.; and Ommer, B. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 10684–10695.

Ruiz, N.; Li, Y.; Jampani, V.; Pritch, Y.; Rubinstein, M.; and Aberman, K. 2023. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 22500–22510.

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. 2022a. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems*, 35: 36479–36494.

Saharia, C.; Ho, J.; Chan, W.; Salimans, T.; Fleet, D. J.; and Norouzi, M. 2022b. Image super-resolution via iterative refinement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(4): 4713–4726.

Salman, H.; Khaddaj, A.; Leclerc, G.; Ilyas, A.; and Madry, A. 2023. Raising the cost of malicious ai-powered image editing. *arXiv preprint arXiv:2302.06588*.

Shang, Y.; Gao, C.; Chen, J.; Jin, D.; Ma, H.; and Li, Y. 2023. Enhancing Adversarial Robustness of Multi-modal Recommendation via Modality Balancing. In *Proceedings of the 31st ACM International Conference on Multimedia*, 6274–6282.

Sharma, G.; and Bala, R. 2017. *Digital color imaging handbook*. CRC press.

Song, J.; Meng, C.; and Ermon, S. 2020. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*.

Song, Y.; and Ermon, S. 2019. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32.

Song, Y.; Sohl-Dickstein, J.; Kingma, D. P.; Kumar, A.; Ermon, S.; and Poole, B. 2020. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*.

Sun, J.; Nie, W.; Yu, Z.; Mao, Z. M.; and Xiao, C. 2022. Pointdp: Diffusion-driven purification against adversarial attacks on 3d point cloud recognition. *arXiv preprint arXiv:2208.09801*.

Suvorov, R.; Logacheva, E.; Mashikhin, A.; Remizova, A.; Ashukha, A.; Silvestrov, A.; Kong, N.; Goka, H.; Park, K.; and Lempitsky, V. 2022. Resolution-robust large mask inpainting with fourier convolutions. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, 2149–2159.

Voynov, A.; Aberman, K.; and Cohen-Or, D. 2022. Sketch-guided text-to-image diffusion models. *arXiv preprint arXiv:2211.13752*.

Xue, H.; Araujo, A.; Hu, B.; and Chen, Y. 2023. Diffusion-Based Adversarial Sample Generation for Improved Stealthiness and Controllability. *arXiv preprint arXiv:2305.16494*.

Yang, R.; Srivastava, P.; and Mandt, S. 2022. Diffusion probabilistic modeling for video generation. *arXiv preprint arXiv:2203.09481*.

Yu, C.; Chen, J.; Wang, Y.; Xue, Y.; and Ma, H. 2023a. Improving Adversarial Robustness Against Universal Patch Attacks Through Feature Norm Suppressing. *IEEE Transactions on Neural Networks and Learning Systems*.

Yu, H.; Chen, J.; Ma, H.; Yu, C.; and Ding, X. 2023b. Defending Against Universal Patch Attacks by Restricting Token Attention in Vision Transformers. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1–5. IEEE.

Zhang, J.; Wu, W.; Huang, J.-t.; Huang, Y.; Wang, W.; Su, Y.; and Lyu, M. R. 2022. Improving adversarial transferability via neuron attribution-based attacks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 14993–15002.

Zhang, J.; Xu, Z.; Cui, S.; Meng, C.; Wu, W.; and Lyu, M. R. 2023. On the Robustness of Latent Diffusion Models. *arXiv preprint arXiv:2306.08257*.

Zhou, B.; Lapedriza, A.; Khosla, A.; Oliva, A.; and Torralba, A. 2017. Places: A 10 million image database for scene recognition. *IEEE transactions on pattern analysis and machine intelligence*, 40(6): 1452–1464.

Zhuang, H.; Zhang, Y.; and Liu, S. 2023. A pilot study of query-free adversarial attack against stable diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2384–2391.