Targeted Attack Improves Protection Against Unauthorized Diffusion Customization

Boyang Zheng[1*] **Chumeng Liang**[2*] **Xiaoyu Wu**[1]

[1] Shanghai Jiao Tong University, [2] University of Southern California

Introduction

Adversarial attack against diffusion model as defense for copyright protection

- Produce **protective noise on images to prevent diffusion customization like LoRA-DreamBooth/SDEdit**
- State-of-the-art performance against Stable Diffusion family

Novel target for **targeted attack outperform previous untargeted attacks**

- Use a **fixed velocity target to attack the diffusion model**
- Introduce controllable ugly pattern to customized images

Robustness & Transferability

- Transferrable across SD family (1.x & 2.x)
- Attack remains robust under many purification methods.

Availability:

- Install upon one click on GitHub release!
- **Optimized to run on most consumer-level Nvidia GPU!**

Method

We select a latent target as the direction for misguiding the diffusion model by optimizing a protective noise η:

$$\min_{\eta} \mathbb{E}_t \mathbb{E}_{z'(t)|z'(0)} \|s_\theta(z'(t), t) - \mathcal{T}\|_2^2$$
$$\text{where } z'(0) = \mathcal{E}(x'), x' = x + \eta, \eta \in [-\zeta, \zeta]$$

Figure 2: Target $\mathcal{T}$ (left) and its corresponding image (right)

Algorithm 1 Attacking with Consistent Errors (ACE)

Input: Image x, diffusion model θ, learning rates α, γ, epoch numbers N, M, K, budget ζ, diffusion training objective in Equation 2, ACE/ACE+ objective function J in Equation 4 & Equation 5.
Output: Adversarial example x'
Initialize $x' \leftarrow x$.
for n from 1 to N **do**
  for m from 1 to M **do**
    $\theta \leftarrow \theta - \gamma \nabla_\theta \mathcal{L}_{LDM}(x', \theta)$
  end for
  for k from 1 to K **do**
    $x' \leftarrow x' - \alpha \nabla_{x'} J$
    $x' \leftarrow \text{clip}(x', x - \zeta, x + \zeta)$
    $x' \leftarrow \text{clip}(x', 0, 255)$
  end for
end for
return x'

$$\begin{cases} \max_{\eta} \mathbb{E}_t \mathbb{E}_{z'(t)|z'(0)} \|s_\theta(z'(t), t) - \nabla_{z'(t)} \log p_t(z'(t)|z'(0))\|_2^2, & \text{(untargeted attack)} \\ \min_{\eta} \mathbb{E}_t \mathbb{E}_{z'(t)|z'(0)} \|s_\theta(z'(t), t) - \mathcal{T}\|_2^2, & \text{(targeted attack)} \end{cases}$$

Why it works?

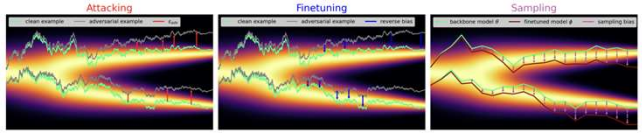

Figure 4: Demonstration of three steps in **Hypothesis 5.1**. First, Attacking step increases ϵ_{adv} of protected images. Second, Finetuning step trains the diffusion model to ϵ_{adv} by a bias $\mathcal{B}_{spl}$, whose direction is reversal to ϵ_{adv}. Third, customized diffusion models include $\mathcal{B}_{spl}$ in sampling so that their output images appear to have chaotic patterns. This hypothesis explains why ϵ_{adv} and $\mathcal{B}_{spl}$ of ACE are reverse to each other as shown in Figure 3.

During attack:
We set a pre-defined latent target as a fixed **direction** to attack diffusion model in training.

During Customization:
The model learned a consistent wrong direction $\mathcal{T}$

Ruing Inference:
The model's velocity prediction will consistently lean to the wrong direction, which compounds over timesteps

[Optional] VAE loss incorporation:
We additionally include the same target to attack the VAE in LDM:

$$\min_{\eta} \mathbb{E}_t \mathbb{E}_{z'(t)|z'(0)} \|s_\theta(z'(t), t) - \mathcal{T}\|_2^2 + \alpha \|z'(0) - \mathcal{T}\|_2^2$$
$$\text{where } z'(0) = \mathcal{E}(x'), x' = x + \eta, \eta \in [-\zeta, \zeta]$$

Visual results

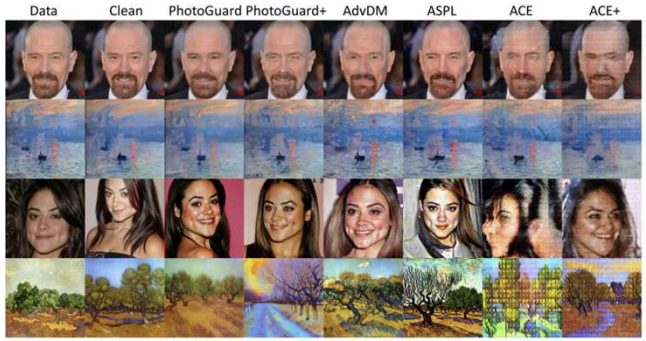

Figure 1: Output images of two mainstream diffusion customization, SDEdit (top two rows) and LoRA (bottom two rows) under different protections with perturbation budget 4/255. **ACE** and **ACE+** are our targeted attack, while others are baselines based on untargeted attacks.

Results

Table 1: Comparison of baseline protections and our protections on LoRA and SDEdit. ACE is our basic method in Equation 4. ACE+ combines ACE and an existing targeted attack (Equation 5). ACE* uses a target other than ACE/ACE+.

	CelebA-HQ					WikiArt				
	SDEdit			LoRA		SDEdit			LoRA	
	MS-SSIM ↓	CLIP-SIM ↓	CLIP-IQA ↑	FDFR ↑	ISM ↓	MS-SSIM ↓	CLIP-SIM ↓	CLIP-IQA ↑		
Clean	0.88	93.38	20.66	0.02	0.69	0.62	89.77	22.88		
PG	0.86	89.24	27.52	0.02	0.72	0.62	88.01	37.52		
PG+	0.82	91.00	22.91	0.04	0.71	0.57	89.80	32.62		
AdvDM	0.81	83.41	24.53	0.04	0.71	0.30	85.29	34.03		
ASPL	0.82	84.12	33.62	0.33	0.48	0.30	87.25	46.74		
ACE	0.73	74.70	31.46	**1.00**	**N/A**	0.23	76.13	40.54		
ACE+	**0.69**	**67.47**	**35.68**	0.07	0.58	0.29	76.07	48.53		
ACE*	0.76	72.52	32.76	0.75	0.47	**0.13**	**73.92**	**76.50**		

Table 2: Transferability results of ACE (top) and ACE+ (middle) and visualization of ACE's output images (bottom). MS, CS, and CI stand for MS-SSIM, CLIP-SIM, and CLIP-IQA. Our methods maintain effective across different models by bringing different degradation to the output images.

Victim	SD1.4			SD1.5			SD2.1		
Backbone	SDEdit		LoRA	SDEdit		LoRA	SDEdit		LoRA
	MS ↓	CS ↓	CI ↑	MS ↓	CS ↓	CI ↑	MS ↓	CS ↓	CI ↑
No attack	0.85	91.71	20.32	0.85	91.16	19.22	0.80	79.00	16.78
SD1.4	0.73	77.24	38.13	0.73	77.58	35.98	0.62	60.82	35.45
SD1.5	0.73	77.29	36.65	0.73	77.50	32.11	0.72	60.10	45.05
SD2.1	0.72	76.20	46.08	0.62	76.80	39.08	0.60	59.12	43.89

Defense	NA	Gaussian		JPEG		Resizing		SR	DiffPure
Params		$\sigma = 4$	$\sigma = 8$	$Q = 20$	$Q = 70$	2x	0.5x		
LoRA									
SDEdit									

Community

Join us on discord, qq group and star on GitHub!