This ICCV paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

When and Where do Data Poisons Attack Textual Inversion?

Jeremy Styborski[1*] Mingzhi Lyu[2*] Jiayou Lu[1] Nupur Kapur[1] Adams Wai-Kin Kong[1]
[1]College of Computing and Data Science, Nanyang Technological University, Singapore
[2]Rapid-Rich Object Search (ROSE) Lab, Nanyang Technological University, Singapore
{styb0001,lyum0002,jiayou001,nupur003,AdamsKong}@ntu.edu.sg

Abstract

Poisoning attacks pose significant challenges to the robustness of diffusion models (DMs). In this paper, we systematically analyze when and where poisoning attacks textual inversion (TI), a widely used personalization technique for DMs. We first introduce Semantic Sensitivity Maps, a novel method for visualizing the influence of poisoning on text embeddings. Second, we identify and experimentally verify that DMs exhibit non-uniform learning behavior across timesteps, focusing on lower-noise samples. Poisoning attacks inherit this bias and inject adversarial signals predominantly at lower timesteps. Lastly, we observe that adversarial signals distract learning away from relevant concept regions within training data, corrupting the TI process. Based on these insights, we propose Safe-Zone Training (SZT), a novel defense mechanism comprised of 3 key components: (1) JPEG compression to weaken high-frequency poison signals, (2) restriction to high timesteps during TI training to avoid adversarial signals at lower timesteps, and (3) loss masking to constrain learning to relevant regions. Extensive experiments across multiple poisoning methods demonstrate that SZT greatly enhances the robustness of TI against all poisoning attacks, improving generative quality beyond prior published defenses.

Code: *www.github.com/JStyborski/Diff_Lab*
Data: *www.github.com/JStyborski/NC10*

1. Introduction

The image quality and prompt fidelity offered by diffusion-based image generation models such as DALL-E 2 [57], Stable Diffusion [58], and Imagen [60] have popularized (or vilified) the use of AI-generated images in the art, marketing, and media industries. The subsequent proliferation of personalization and editing methods [15, 20, 25, 35, 59, 82] and tools [2, 7, 72] allows anyone, even non-experts and non-artists, to quickly retrain existing models to generate their desired images, including images of novel concepts not found in the model training data.

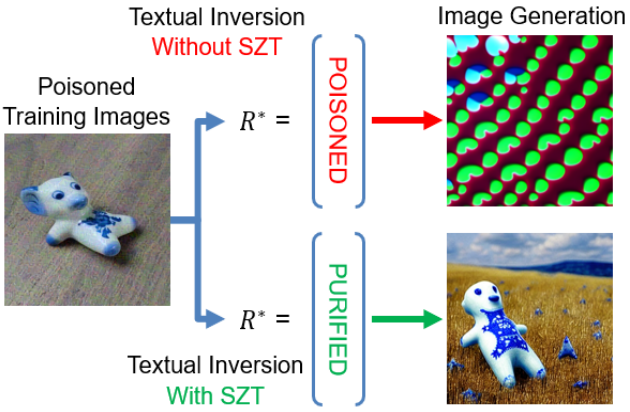


Figure 1. Applying SZT to TI mitigates poison signals and recovers shape, color, and texture of desired concepts. NovelConcepts10 training data poisoned by ADM+. Generation prompt: "A $R*$ with a wheat field in the background"

Unfortunately, the surge in generative AI corresponds to misfortune for artists and copyright owners, who may see their works easily reproduced by retrained generative models. Consequently, the subfield of data poisoning for generative AI has focused on how to inject images with signals that nullify their use in training datasets for generative models. Individuals looking to protect their data are encouraged to "poison" their images with imperceptible adversarial perturbations such that generative models will fail to learn to recreate the novel concepts contained within.

The concept of data poisoning stems from adversarial examples (AEs) [17, 36, 49, 70], where the inputs are slightly perturbed in order to maximize loss. Intuitively, these AEs learn to incorporate signals that maximally mislead AI models into making incorrect predictions. The AE method is likewise effective against generative models; multiple state-of-the-art poisoning methods for generative models apply the AE method to the diffusion model (DM) objective to inject poison signals into images [6, 39, 61, 78]. Although the textual inversion (TI) technique is widely used to adapt DMs to novel concepts, there are no systematic studies that identify when and where AEs attack the TI process.

* Equal contribution, may cite either first

1.1. Research Gap and Contributions

State-of-the-art poisoning methods for DMs are largely cannibalizing previous adversarial methods (e.g., Glaze [63] from Adv-VAE [71] or AdvDM [39] from PGD [36, 49]) or creating new loss functions that nonetheless follow the same AE algorithm (e.g., SDS [78] or DiffusionGuard [6]). Although these poisoning methods are useful research milestones that explore novel applications for AEs, they neglect to study how poison signals are learned and which DM vulnerabilities are being exploited. Furthermore, many poisoning works neglect to understand the intricacies of the downstream tasks, such as TI, for which the AEs are designed.

In this paper, we examine the learning behavior of TI [15], a lightweight personalization technique for adapting DMs to novel concepts, when trained on poisoned datasets. To enable our analysis, we devise a novel visualization method, semantic sensitivity maps (SSM), to display loss sensitivity to individual text tokens throughout training. Using SSMs, we observe several heretofore unseen trends in poison learning: (1) DMs concentrate poison learning at lower-middle timesteps (approximately $t \in [0, 600]$) and (2) adversarial signals tend to "distract" learning away from the novel concept of interest and towards irrelevant regions.

We scrutinize the noising process and the noise-prediction training objective that are common to DMs and the TI process to demonstrate that their combined effect inherently favors low-noise images. Crucially, this learning bias towards lower-middle timesteps is inherent in most AE algorithms for DMs, which implies that the AEs themselves are mostly learned at lower-middle timesteps. This exposes a weakness where many poisoning techniques are less effective at higher timesteps. In experiments, we demonstrate that restricting TI training to higher timesteps is a simple yet effective defense against multiple poisons.

We find that TI prioritizes learning in regions that contain concept objects rather than background objects. This behavior stems from the tendency to minimize noise-prediction loss by focusing on common features across training images. Poisoning disrupts this process by introducing dominant adversarial signals and suppressing the learnability of clean concept information, misleading the TI training process. To counteract this, we propose a loss-masking strategy during optimization, explicitly training the model to focus on the concept regions within poisoned images. This ensures that TI effectively captures the intended concept and mitigates the influence of adversarial signals.

We also analyze JPEG compression, a common defense technique in poisoning literature. We find that JPEG compression helps purify AEs by reshaping the distributions of adversarial signal magnitudes and frequencies to benign distributions, similar to those of clean images. Although JPEG compression alone cannot enable TI to learn from poisoned images, using it as a first step to weaken high-frequency adversarial signals significantly improves the performance of other defense methods.

Finally, we combine our analyses to present safe-zone training (SZT), a simple and lightweight defense method for training TI. SZT consists of three main components: (1) JPEG compression to weaken high-frequency adversarial signals, (2) training at high timesteps to avoid adversarial signals at lower-middle timesteps, and (3) loss masking to prioritize learning on the novel concept. We show that SZT outperforms multiple established defenses across six different poisons in accurately generating novel concepts and reaches the performance of TI on clean data.

Our contributions are summarized below:

- We devise SSMs and use them to visualize loss attribution during the TI training process.
- We analyze the noising process and noise-prediction objective to understand the mechanism by which DMs, AEs, and TI focus on lower-middle timesteps.
- We develop a loss-masking strategy to encourage DMs to learn from the concept regions in the training data.
- We combine our findings and establish SZT to methodically improve the poison robustness of TI.

2. Related Works

We assume that the reader is familiar with standard DMs [24, 32, 67, 68] and generation techniques [12, 23, 50, 66]. We mainly focus on latent diffusion models (LDMs) [58].

Although DMs can easily generate most concepts from their training dataset with high quality, they predictably suffer in generation quality for novel (unseen) concepts. Editing techniques [3, 20, 29, 50–52, 82] apply modifications to an input image while enforcing structural or composition constraints, permitting generation with novel concept images. However, editing techniques are still bound to the prior from the DM's training data. In order to "inject" new concepts and styles into the DM, personalization techniques [15, 18, 19, 25, 35, 53, 59, 73, 74, 83] quickly finetune some component of the DM on a small dataset of novel concept images. Many personalization techniques learn new text embeddings [15, 18, 35, 73, 74] or co-opt existing tokens [19, 59] to capture concept semantics. Without retraining the entire DM from scratch, personalization techniques are the best way to adapt DMs to new concepts. In particular, TI [15] is a lightweight personalization method that requires training only one new text embedding and is often included within other methods [18, 35, 73].

AEs [5, 17, 36, 49, 70] expose the brittleness of neural net predictions by learning minor perturbations to maximize loss. Although originally designed for classification models, AEs have also been applied to object detection and segmentation [76], reinforcement learning [27], watermarking [48], self-supervised learning [4, 28, 30], VAEs [34, 71], and diffusion models [6, 37, 39, 61, 63, 64, 78]. Due to their

ability to defeat deep learning models, AEs have become a landmark technique for data protection. The concept of AEs extends to adversarial poisoning [14, 26], where entire datasets are poisoned to nullify attempts to scrape or steal data. Typical defenses against AEs include simple detection [1], augmentation [8, 9, 11, 45, 80, 81], which modifies input images to increase sample diversity and dilute poison signals, multitask architectures [69, 79], which use multiple networks to mitigate shortcut learning [16, 47], and adversarial training, which simply adds AEs to the training set.

AEs for DMs fall into two main categories. Encoder attacks [61, 63, 64] utilize the encoder of an LDM to learn a small perturbation to an input image that minimizes the distance between the perturbed image encoding and the encoding of some unrelated target image. Diffuser attacks [6, 37, 39, 61, 78] aim to learn a small perturbation to an input image that maximizes the DM loss (often noise-prediction). Other attack methods combine encoder attacks and diffuser attacks [38, 61]. Defenses against AEs for DMs include augmentation [65] and regeneration [13, 44, 77, 84], where poisoned images are randomly noised and then denoised with a DM to remove adversarial signals. Regeneration methods are the most actively researched, but they are time-consuming and complex as they require preprocessing entire datasets with a DM in order to "purify" them.

3. Preliminaries

3.1. Latent Diffusion Models

We mainly focus on LDMs [58] as they are quick to train, they generate high-quality outputs, and tools for using/modifying them are widely available. In LDMs, an input image x is first encoded to a latent of lower dimension via a variational encoder, $z_0 = \mathcal{E}(x)$[1]. The latent is then iteratively noised to timestep t according to noise variance schedule β_t. An equivalent one-step noising process is

$$z_t = \sqrt{\overline{\alpha}_t} z_0 + \sqrt{1 - \overline{\alpha}_t}\epsilon, \tag{1}$$

where $\overline{\alpha}_t = \Sigma_{s=1}^{t}(1 - \beta_s)$ and $\epsilon \sim \mathcal{N}(\mathbf{0}, I)$. $\overline{\alpha}_t$ monotonically decreases from 1 to 0 as t increases from 0 to T, following a variance-preserving schedule [68]. A generative DM learns the denoising process, where the goal is to maximize the likelihood of generating the unnoised input (z_0). DDPM [24] showed that this objective can be practically realized with a simple noise-prediction task. The model predicts the Gaussian noise ϵ added to input z_0 when given noisy sample z_t, and the learning objective is given as

$$\arg\min_{\theta} E_{x,t\sim\mathcal{U}(0,T),\epsilon}\left[||\epsilon_\theta(z_t, t) - \epsilon||_2^2\right], \tag{2}$$

where x is drawn from the training dataset, $\mathcal{U}(0, T)$ is the uniform distribution between timesteps 0 and T, and ϵ_θ is a model that predicts a noise vector ϵ given a noised latent z_t. For text-conditional models, $\epsilon_\theta(z_t, t, \tau_\theta(y))$ also accepts features vector $\tau_\theta(y)$, where τ_θ is a language model and input prompt y is provided by the user or training data.

[1] We abuse notation and let $\mathcal{E}(x)$ encompass encoding and sampling.

3.2. Textual Inversion

In this paper, we consider how AEs affect the downstream personalization task of TI [15], which seeks to learn a new text token that captures the semantics of a novel concept. TI trains a new text embedding corresponding to a new token R^*; the new embedding is the only finetuned parameter. TI trains via a noise-prediction task as

$$\arg\min_{\theta_{R^*}} E_{x_f,t\sim\mathcal{U}(0,T),y^*,\epsilon}\left[||\epsilon_\theta(z_t, t, \tau_\theta(y^*)) - \epsilon||_2^2\right], \tag{3}$$

where sample x_f is drawn from the finetuning dataset and y^* is a prompt containing token R^*.

3.3. Adversarial Examples

The primary method to create AEs against LDMs is by attacking the diffuser with an adversarial perturbation δ injected into the input data. The adversarial objective is to find a δ that maximizes the noise-prediction loss, as

$$\begin{aligned} \arg\max_{\delta} & E_{x_p,t\sim\mathcal{U}(0,T),\epsilon}\left[||\epsilon_\theta(z_t', t) - \epsilon||_2^2\right] \\ & \text{s.t.} \quad ||\delta||_b \leq \kappa, \end{aligned} \tag{4}$$

where sample x_p is drawn from the dataset to poison and $z_t' = \sqrt{\overline{\alpha}_t}\mathcal{E}(x_p+\delta)+\sqrt{1-\overline{\alpha}_t}\epsilon$ is the adversarial latent following Eq. 1. The perturbation δ is commonly constrained in the l^∞ norm to a small value κ to limit its visibility.

Encoder attacks against LDMs seek to perturb an image such that the encoding of the poisoned image $x_p + \delta$ is similar to that of some unrelated target image x_g. Encoder attacks optimize the objective function

$$\arg\min_{\delta} E_{x_p}\left[||\mathcal{E}(x_p + \delta) - \mathcal{E}(x_g)||_2^2\right] \text{ s.t. } ||\delta||_b \leq \kappa. \tag{5}$$

4. Analysis

4.1. Visualizing the Effect of Adversarial Examples

To observe the influence of AEs against TI, it is critical to visualize the impact of each text embedding on a generated image. Let $\widehat{e} = \{e_0, ...e_n, ..., e_{L-1}\}$, where e_0 represents the start token embedding vector and e_n represents the embedding vector of the nth token. For a text input consisting of l tokens, $\widehat{e}$ is padded to length L with $\{e_{l+1}, ..., e_{L-1}\}$ with end token embeddings. Each e_n is obtained via lookup in a token-embedding dictionary. As an example, given the text prompt, "a puppy wearing a hat", embeddings $\{e_1, ..., e_5\}$ correspond to "a", "puppy", "wearing", "a", and "hat" respectively.

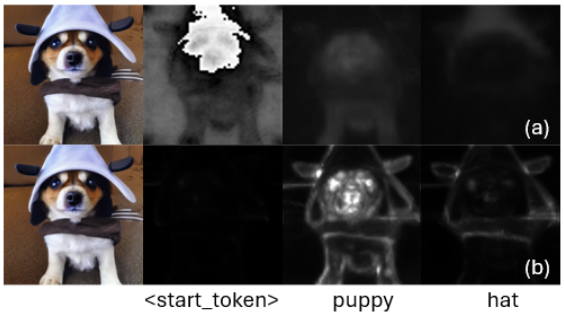


Figure 2. Cross attention maps (a) and SSMs (b) for tokens "<start_token>", "puppy", and "hat" with text prompt "a puppy wearing a hat" at noise timestep $t = 500$.

A popular text-attribution method for LDMs visualizes the cross-attention maps [41] in ϵ_θ. However, cross-attention maps are incapable of capturing the effect of each embedding e_n. This is because the self-attention layers in the text encoder τ_θ entangle signals from the embedding vectors of $\widehat{e}$ before they reach the cross-attention layers of ϵ_θ. Fig. 2(a) visualizes text attribution using cross-attention maps. The cross-attention map corresponding to the start token e_0 is bright, highlighting many regions, but the maps corresponding to words "puppy" and "hat" are dim and blurry. This drawback has been noted in other works [46].

To accurately identify the spatial regions over which a specific token has influence, we developed a new visualization method called semantic sensitivity maps (SSMs). We define SSMs by

$$SSM(x,t,\widehat{e},n) = E_{\widehat{e}_{\Delta n}}\left[||\epsilon_\theta(z_t,t,c) - \epsilon_\theta(z_t,t,c_{\Delta n})||^2_{ch}\right]$$
$$\text{s.t. } c = \tau_\theta(\widehat{e}),\ c_{\Delta n} = \tau_\theta(\widehat{e}_{\Delta n}), \quad (6)$$

where x is an input image and c is the condition tensor. $c_{\Delta n}$ is analogous to c, except that it is derived from $\widehat{e}_{\Delta n}$, a modified version of $\widehat{e}$ that replaces e_n with a randomly sampled vector from the embedding dictionary. For notational simplicity, we denote that $\tau_\theta(\widehat{e})$ accepts token embeddings as input instead of the prompt. $||\cdot||^2_{ch}$ indicates sum of squares across channels such that the output is a one-channel image that matches the input height and width. Fig. 2(b) displays the SSMs corresponding to the embeddings for "puppy" and "hat". Compared to cross-attention attribution maps in Fig. 2(a), SSMs more accurately reflect the regions corresponding to each input embedding. Crucially, SSMs avoid attributing semantic tokens to the start token.

Using SSMs, we can investigate when and where adversarial signals attack the TI process. Fig. 3 shows the SSMs for a clean image and its corresponding AEs for various noising timesteps. The input images contain a novel concept not contained in the training dataset. The image prompt is "a R^* laying on top of a grass covered field". Fig. 3 shows the SSMs for the trainable text embedding corresponding to R^* before and after TI training. We observe

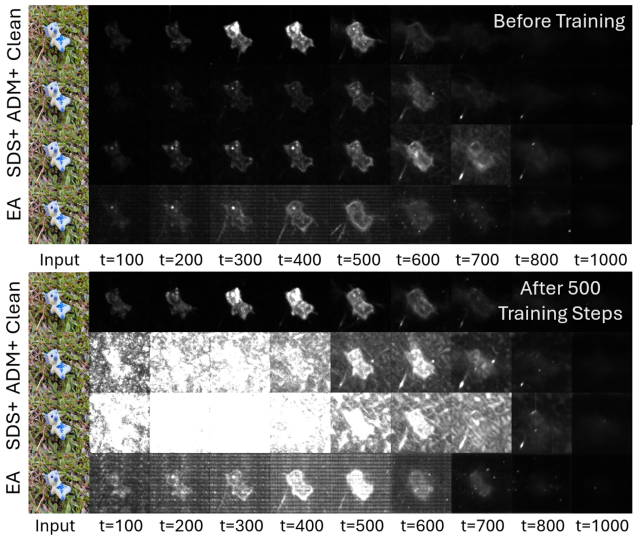


Figure 3. Comparison between SSMs of clean and AE samples at different noise timesteps (a) before TI training and (b) after 500 steps of TI training.

that before TI training, SSMs of the clean sample and AEs highlight the novel object in the range $t \in [300, 700]$. This indicates that the DM prior can already correlate unknown objects with unknown tokens. After TI training, the novel concept is more pronounced in SSMs of clean images, but the SSMs of AEs become extremely noisy, particularly at lower-middle timestep values. Notably, the attribution for AEs extends far outside the novel concept region. To explain this behavior, we investigate the temporal and spatial learning properties of TI in subsequent sections.

4.2. Timestep Learning Bias

Fig. 3 shows that concept learning is focused on lower-middle timesteps ($t < 700$) and that poison signals have the greatest impact at lower timesteps. To understand this phenomenon, we revisit the noising and training processes.

We begin by noting that all training objectives (i.e., model training in Eq. 2, TI in Eq. 3, or AEs in Eq. 4) utilize the expected noise-prediction loss over a uniform t distribution. The only differences between the objectives are the training datasets and the trainable parameters.

Recalling that $\overline{\alpha}_t$ from Eq. 1 monotonically decreases from 1 at $t = 0$ to 0 at $t = T$, we can see that the noising process is interpolating between input z_0 and noise ϵ. At $t = 0$, $z_{t=0} = \sqrt{\overline{\alpha}_0}z_0 + \sqrt{1-\overline{\alpha}_0}\epsilon = z_0$. Therefore, at $t = 0$, the input to the noise-prediction model $\epsilon_\theta(z_0, 0)$ contains no information about the sampled noise ϵ. Since ϵ is distributed as $\mathcal{N}(\mathbf{0}, I)$, the optimal noise prediction to minimize the expected loss at $t = 0$ is $\epsilon_\theta(z_0, 0) = \mathbf{0}$. Predictably, the loss at $t = 0$ is a maximum because the noise-prediction model cannot accurately predict the ground-truth noise. Conversely, at $t = T$, $z_{t=T} = \sqrt{\overline{\alpha}_T}z_0 + \sqrt{1-\overline{\alpha}_T}\epsilon = \epsilon$, and the

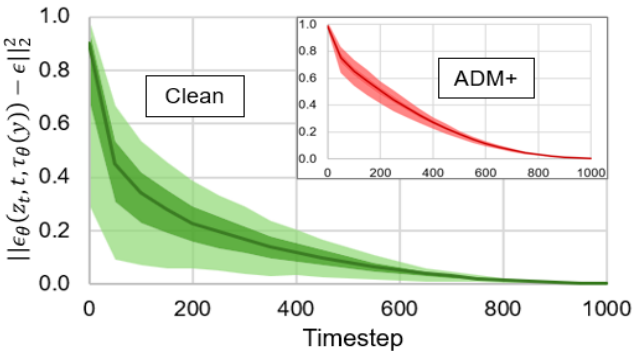


Figure 4. Loss distributions for Stable Diffusion 1.5 on 1000 clean (green) and 1000 ADM+ poisoned (red) LAION Aesthetic images at each of 21 interspersed timesteps. The dark lines represent median loss values. The dark and light bands encompass 50% and 95%, respectively, of the loss distributions.

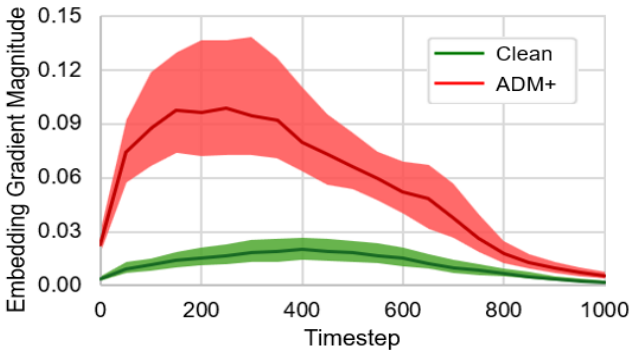


Figure 5. Trainable text embedding loss gradients after 1000 steps of TI training on clean and ADM+ poisoned NovelConcepts10 images. Results at each timestep are averaged across all concepts. The dark lines represent median gradient values. The dark bands encompass 50% of the gradient distributions.

noise-prediction model simply needs to return the input to minimize loss. At $t = T$, the loss approaches a minimum.

To validate our argument, we calculate the noise-prediction loss of Stable Diffusion 1.5 for 1000 images sampled from the LAION Aesthetic [62] dataset at 21 equally-spaced timesteps (21000 samples total) and plot the resulting distributions in Fig. 4. As predicted, the expected loss decreases from a maximum at $t = 0$ to 0 at $t = T$. Applying ADM+ [39] poisoning to the images increases the loss values in the lower-middle timesteps (as expected by a loss-maximizing AE algorithm), but the losses at $t = 0$ and at high timesteps remain unchanged.

We must also consider how the noise-prediction objective affects gradients during TI training. Fig. 5 displays the average loss gradient magnitude with respect to the trainable text embedding at each timestep after 1000 steps of TI training with Stable Diffusion 1.5. Curves are obtained by calculating 1000 losses for each concept of the NovelConcepts10 dataset across 21 equally-spaced timesteps and then averaging gradient magnitudes at each timestep across concepts. As expected, the graph shows that at high timesteps, gradients decay to 0 as the loss distributions converge to 0. At lower-middle timesteps, moderate loss allows the model to backpropagate real (or adversarial) signals to the text embedding, corresponding to high gradients. However, near $t = 0$, high losses correspond to low gradients. Intuitively, this is because the noise-prediction model minimizes loss at $t = 0$ by predicting near-$\mathbf{0}$ noise regardless of the inputs, z_t and $\tau_\theta(y^*)$. Since the prediction is independent of the inputs, no loss signal is backpropagated to the trainable text embedding. The loss gradient magnitudes in Fig. 5 correspond to the intensities of the sensitivity maps in Fig. 3.

This behavior is particularly applicable to diffusion-based AEs and TI. All diffusion-based AEs seek to maximize the expected noise-prediction loss (Eq. 2), which includes an expectation over a uniform timestep range $\mathcal{U}(0, T)$. As the loss gradient is near $\mathbf{0}$ at high timesteps, most of the adversarial signals will be learned at lower-middle timesteps. Likewise, TI applies an expectation over a uniform timestep range (Eq. 3) thus focusing learning at lower-middle timesteps. Assuming that the adversarial signals contained within the training images are biased towards lower-middle timesteps, then we can bias the TI task towards high timesteps to avoid adversarial signals and learn clean concept features.

In Appendix B, we give a detailed derivation of the noise-prediction error along with additional plots for the loss and noise-prediction distributions. In Appendix C, we show the resulting effect on loss gradients throughout TI training and across multiple poisons.

4.3. Spatial Learning Bias

As observed in Sec. 4.1, AEs disrupt TI by "distracting" the optimization process and expanding the spatial distribution of the trainable token attribution beyond the novel concept region. To understand this phenomenon, we revisit the adversarial objectives in Eq. 4 and Eq. 5.

We begin by noting that both adversarial objectives involve an l^2 norm which includes a sum over all spatial coordinates (height, width, channels) of the error tensor. This summation does not apply preferential weight to any spatial region. Furthermore, the l^2 vector norm is convex and non-decreasing, implying that it increases as the magnitudes of individual elements increase. This encourages all spatial elements to contribute to optimizing the l^2 objective. This incentive combines with the adversarial perturbation constraint $||\delta||_b$, which is commonly implemented with the l^∞ norm. We note that $||\delta||_\infty$ is non-increasing for all infinitesimal changes in perturbation elements δ_i where $|\delta_i| < ||\delta||_\infty$; this permits increases in most perturbation elements without affecting the constraint. In summary, the l^2 norm objectives in Eq. 4 and Eq. 5 encourage all elements

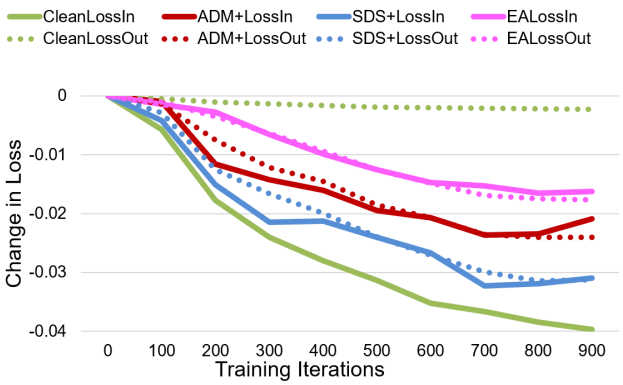


Figure 6. Loss difference inside ("LossIn") and outside ("LossOut") novel concept regions during TI training on clean, ADM+, SDS+, and EA images of seven concepts from NovelConcepts10.

of the error terms to contribute and the l^∞ constraint on δ encourages all elements of δ to contribute maximally. Practically, this implies that adversarial perturbations maximally infect every region of a poisoned image.

To empirically verify this phenomenon, we study the losses attributed to novel concept regions for clean images and AEs during TI training. We apply TI to clean, ADM+, SDS+, and EA versions of the NovelConcepts10 dataset. We track the loss contributions from inside and outside the novel concept regions throughout training. We first average losses across their contributing pixels and then average the results across all concepts. The results in Fig. 6 indicate that TI for clean images can effectively minimize the loss inside the novel concept regions and ignores signals outside these regions. This implies that TI effectively learns new concepts and ignores background information. However, when trained on AEs, TI losses inside and outside of the concept regions are reduced by the same proportion. This indicates that AEs mislead TI to learn equally from all regions and ignore the semantic significance of the novel concept. This aligns with the observation from Fig. 3 that all regions of an adversarial input are attributed to the trainable token.

4.4. JPEG as Poison Defense

The adversarial signals injected into images by poisoning techniques tend towards higher frequencies in pixel space. Consequently, JPEG compression, which relies on discrete cosine transforms to identify and remove high-frequency signals, has been noted as an effective poison defense [45]. We investigate the mechanism by which JPEG eliminates adversarial signals and identify two crucial behaviors: (1) JPEG compression converts bimodal poison noise into unimodal Gaussian-like noise in pixel space and (2) JPEG centralizes the frequency spectrum in latent space, forcing poisoned latent frequencies towards those of clean latents.

We empirically verify (1) in Fig. 7 by examining the distribution of poison perturbations (relative to clean images) with and without applying JPEG compression. We use a JPEG quality of 25 and aggregate results across all 50 images in the NovelConcepts10 dataset. The bimodal distribution of poison perturbations aligns with loss objective and constraint discussion from Sec. 4.3. After JPEG compression, the poison perturbations follow a unimodal Gaussian-like distribution. We note in Fig. 12 of Appendix D that this behavior holds for all poisons studied. Intuitively, the carefully-optimized poison signals have been replaced by a distribution of random errors due to JPEG compression.

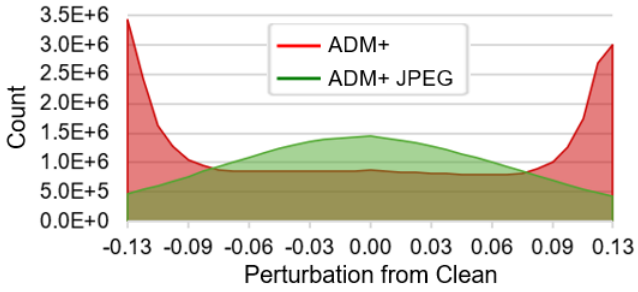


Figure 7. Histograms of pixel-space perturbations (relative to clean images) for ADM+ poisoned images without (red) and with (green) JPEG compression.

Due to space limitations, we move empirical verification of (2) and subsequent analysis to Appendix D. We note that the effect of (2) is that JPEG-compressed poisoned images force poison latents to conform to the power spectra of natural image latents, as would be expected by LDMs.

4.5. Safe-Zone Training

In the above analyses, we have demonstrated when and where AEs poison TI. Combining takeaways from our analyses, we propose Safe-Zone Training (SZT) to mitigate the influence of adversarial signals in poisoned images. Applying lessons from Sec. 4.4, we first apply JPEG compression to obfuscate the adversarial signals in the pixel domain and centralize the power spectra in the latent domain. Applying the conclusions from Sec. 4.2, we restrict training to high timesteps in order to avoid adversarial signals at lower-middle timesteps. Finally, SZT incorporates the behaviors noted in Sec. 4.3 and utilizes a binary loss mask to allow only signals from the novel concept region to backpropagate to the trainable text embedding. In summary, SZT is a simple algorithm that involves applying JPEG compression, adjusting the timestep sampling range, and inserting a mask during loss calculation. Crucially, SZT does not require additional networks to regenerate or purify poison data. The SZT objective can be formulated as

$$\arg\min_{\theta_{R^*}} E_{x'_j,t,y^*,\epsilon}\left[||(\epsilon_\theta(z'_t,t,\tau_\theta(y^*))-\epsilon)\odot M_z||_2^2\right] \quad (7)$$

$$\text{s.t. } x'_j = JPEG(x'_p),\ t\sim\mathcal{U}(t_{th},T),\ M_z = Resize(M_x),$$

where x'_j is the JPEG-compressed version of poisoned image $x'_p = x_p + \delta$, $z'_t = \sqrt{\overline{\alpha}_t}\mathcal{E}(x'_j) + \sqrt{1-\overline{\alpha}_t}\epsilon$, timestep t is uniformly sampled from the range above threshold t_{th}, M_z is the latent-space mask resized from the pixel-space mask of the novel concept region M_x, and $\odot$ denotes element-wise multiplication.

5. Experiments

5.1. Experiment Settings

5.1.1. Datasets

We test the ability of TI to learn new concepts from poisoned images. To ensure that the LDM is learning new concepts not contained within the prior, we collected NovelConcepts10, a dataset of five images for each of 10 distinct objects (50 total images). An extended dataset description and example images are given in Appendix E. We also evaluate on CustomConcept101 [35], which was curated to study personalization tasks. For each image in a dataset, we create a binary mask of the novel concept region using the Segment Anything Model [33]. We utilize six data poisons: ADM+ [39], ADM-, SDS+ [78], SDS- [78], EA [61], and DA [61]. We place poisoning details in Appendix F.

5.1.2. Model and Training

We conduct all experiments with Stable Diffusion v1.5 [58]. We utilize a modified Diffusers library from HuggingFace [72] to perform TI. For each concept of a dataset, we run TI [15] with a default learning rate of $5e^{-4}$ (no decay, no warmup) for 5000 training steps and a batch size of 1. For CustomConcept101, we utilize 2500 training steps without noticeable loss of performance. For each learned concept, we generate five images for each of the 24 object prompts from the DreamBooth pipeline [59] (120 images total).

5.1.3. Evaluation Metrics

For each learned concept, we measure the DINOv2 encoding similarity [54] and FID score [22] between training images and the images generated after TI. For poisoned datasets, we calculate metrics between generated images and clean versions of the training images. We also measure prompt fidelity with CLIP Score [21, 56] between the generation prompt and the generated images. We report metrics first averaged across all generated images per concept and then across all concepts in a dataset.

5.2. Results and Analysis

5.2.1. Personalization Settings Ablations

We begin by ablating training hyperparameters for TI across clean, ADM+, ADM-, and EA versions of NovelConcepts10 for learning rate in ($5e^{-5}$, $5e^{-4}$, $5e^{-3}$), learning rate schedule in (constant, linear decay, cosine decay), and training steps in (1000, 5000, 10000). Results and further discussion are given in Appendix G. In summary, we find that maintaining default settings (constant $5e^{-4}$ learning rate with 5000 training steps) is sufficient to learn novel concepts without overfitting to the training images.

5.2.2. Timestep Range Ablation

Based on findings about timestep learning bias in Sec. 4.2, we ablate methods to restrict timesteps to high values during TI training. We consider multiple sampling methods to favor high timesteps: thresholding, power distributions, and $tanh$ distributions. Fig. 15 in Appendix H displays the probability curves for the power and $tanh$ distributions. We ablate t-sampling methods for clean, ADM+, ADM-, and EA versions of NovelConcepts10. Full results are given in Appendix H. We find that simple high-thresholding works best, improving DINOv2 similarity for ADM+ data from 0.10 ($t \sim \mathcal{U}(0, 1000)$) to 0.21 for $t \geq 600$ and 0.28 for $t \geq 700$. Using $t \geq 600$ also maintains or improves scores for clean and EA datasets. In our subsequent proposed defenses, we mainly utilize $t \geq 600$, denoted "T600".

5.2.3. Masking Ablation

To verify the effectiveness of loss masking (LM), we compare it to various masking strategies: input image masking (IM), latent (z_t) masking (ZM), and masking both input image and loss (LIM). We ablate the masking method on all poisoned versions of NovelConcepts10. Results in Appendix I demonstrate that LM outperforms all other masking strategies across all poisoning methods, except EA. We hypothesize that applying masks directly to the input image or latent z_t may destroy useful information before the features are processed by the U-Net. Intuitively, masked inputs lack contextual information about the relationship between the novel concept and the background, distorting the U-Net predictions. By applying the mask to the final loss, we preserve background information and permit a more coherent understanding of the scene.

In Appendix J, we experiment with dilating the mask size to incorporate additional context in loss backpropagation. We find that relatively small amounts of mask dilation (i.e., 16 pixels for 512x512 masks) can improve LM for TI.

5.2.4. Safe-Zone Training Performance

Our implementation of SZT is highly customizable, empowering the user to control JPEG compression quality, timestep restriction method/shape, and loss mask size/shape. In our ablations, we find that combining JPEG compression (quality$=25$), timestep restriction $t \geq 500$, and LM with 16 pixels of mask dilation performs best. We defer further discussion on SZT configurations to Appendix J and encourage users to explore additional settings.

We demonstrate the robustness of SZT across poisoned versions of the CustomConcept101 dataset. Table 1 compares the DINOv2 similarity performance of SZT and its novel components with established defenses, including Regen [84], PDMPure [77], AdvClean [65], and JPEG com-

Table 1. CustomConcept101 DINOv2 Similarity for various poison defenses.

Defense → Poison ↓	Nominal	Regen [84]	PDMPure [77]	AdvClean [65]	JPEG	T600	LM	JPEG +T600	JPEG +LM	SZT
Clean	0.46	0.46	0.25	0.45	0.45	**0.50**	0.47	0.47	0.46	0.48
ADM+[39]	0.10	0.17	0.23	0.15	0.36	0.25	0.30	0.41	0.44	**0.45**
ADM-	0.33	0.25	0.23	0.35	0.41	0.27	0.43	0.45	0.45	**0.46**
SDS+[78]	0.10	0.16	0.25	0.16	0.34	0.23	0.32	0.40	**0.45**	0.44
SDS-[78]	0.31	0.22	0.23	0.35	0.40	0.23	0.42	0.45	0.45	**0.47**
EA[61]	0.10	0.09	0.24	0.19	0.33	0.12	0.29	0.44	0.44	**0.46**
DA[61]	0.29	0.25	0.27	0.35	0.39	0.22	0.42	0.44	0.46	**0.47**
Psn Avg	0.21	0.19	0.24	0.26	0.37	0.22	0.36	0.43	0.45	**0.46**

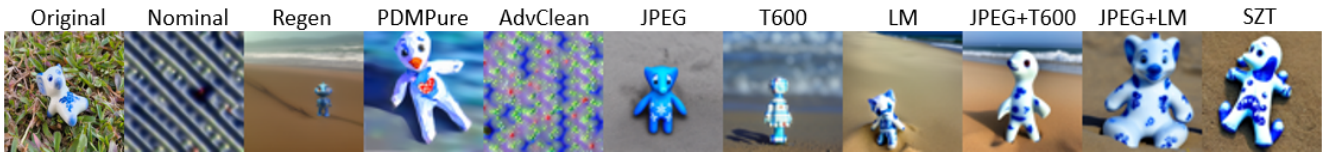


Figure 8. Generated images after TI on NovelConcepts10 BlueElephant poisoned by ADM+. Generation prompt: "a $R*$ on the beach"

pression; we apply Regen default settings with Stable Diffusion 1.5, utilize published repositories for PDMPure and AdvClean, and set JPEG quality to 25.

JPEG is a moderately-strong defense, outperforming established methods like Regen and PDMPure on multiple poisons. T600, despite requiring only one parameter change and no extra computation overhead, modestly improves performance against adversarial poisons like ADM+ and SDS+ and outperforms some established defenses on these poisons. LM is a strong standalone defense, improving the generative quality for every poison. Adding JPEG compression as a preprocessing step for T600 or LM further improves performance across all poisons. In particular, JPEG+T600 is an incredibly efficient defense, delivering competitive image quality across all poisons without additional networks or masking. SZT is consistently the strongest defense, achieving state-of-the-art image quality across almost all poisons. Notably, SZT performance on all poison data (average of 0.46) matches the quality achieved by nominal TI on clean data, effectively nullifying all poisons. We also report FID and CLIP Score for CustomConcept101 in Appendix K. We perform the same study on NovelConcepts10 in Appendix L and find that the trends are largely similar.

We include studies on additional LDM architectures and parameterizations in Appendix O and additional personalization methods in Appendix P. Across all models and methods, SZT and its ablations are the strongest defenses.

For a qualitative comparison, we display images generated after TI training on the BlueElephant concept poisoned by ADM+ in Fig. 8. Existing defenses (Regen, PDMPure, and AdvClean) can occasionally improve generated image quality, resulting in generated images that contain a semblance of the desired concept. Images generated after JPEG compression display some faithfulness to the desired concept. T600 also presents as a moderately effective defense, often removing poison noise but occasionally failing to capture the desired concept. LM alone tends to learn the correct concept shape but not the correct texture or color, as LM does not attempt to mitigate adversarial signals within the masked region. JPEG+LM improves on this weakness. JPEG+T600 strikes a good balance between accurately learning the desired concept and displaying prompt context. SZT is able to accurately learn the desired concept shape and texture. We note that defenses utilizing the LM operation (i.e., LM, JPEG+LM, and SZT) tend to hyperfocus on the concept, generating images that are dominated by the concept. This behavior is reduced for SZT, which uses dilated masks to incorporate some background information from training images. We include further qualitative results for additional poisons and concepts from CustomConcept101 and NovelConcepts10 in Appendix M.

6. Conclusion

The SZT defense method is derived from systematic analyses of the underlying mechanisms of DMs as well as observations on the learning tendencies of TI. Despite the simplicity of SZT, it is an effective defense method that performs beyond other established defenses and nullifies many poisons. We hope that our work exposes the vulnerabilities of existing poisons and spurs further research on robust poisons for DMs.

Acknowledgments

This research is supported by the National Research Foundation, Singapore and Infocomm Media Development Authority under its Trust Tech Funding Initiative, and ROSE @ NTU. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not reflect the views of National Research Foundation, Singapore and Infocomm Media Development Authority, or ROSE @ NTU.

References

[1] Ahmed Aldahdooh, Wassim Hamidouche, Sid Ahmed Fezza, and Olivier Déforges. Adversarial example detection for dnn models: a review and experimental comparison. *Artificial Intelligence Review*, 55(6):4403–4462, 2022. 3

[2] AUTOMATIC1111. Stable Diffusion Web UI, 2022. 1

[3] Tim Brooks, Aleksander Holynski, and Alexei A. Efros. Instructpix2pix: Learning to follow image editing instructions, 2023. 2

[4] Nicholas Carlini and Andreas Terzis. Poisoning and backdooring contrastive learning. In *International Conference on Learning Representations*, 2022. 2

[5] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks, 2017. 2

[6] June Suk Choi, Kyungmin Lee, Jongheon Jeong, Saining Xie, Jinwoo Shin, and Kimin Lee. Diffusionguard: A robust defense against malicious diffusion-based image editing. In *The Thirteenth International Conference on Learning Representations*, 2025. 1, 2, 3

[7] CivitAI, 2025. 1

[8] Jeremy M Cohen, Elan Rosenfeld, and J. Zico Kolter. Certified adversarial robustness via randomized smoothing, 2019. 3

[9] Ekin D. Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V. Le. Randaugment: Practical automated data augmentation with a reduced search space, 2019. 3

[10] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009. 14

[11] Terrance DeVries and Graham W. Taylor. Improved regularization of convolutional neural networks with cutout, 2017. 3

[12] Prafulla Dhariwal and Alex Nichol. Diffusion models beat gans on image synthesis, 2021. 2

[13] Hadi M. Dolatabadi, Sarah Erfani, and Christopher Leckie. The devil's advocate: Shattering the illusion of unexploitable data using diffusion models, 2024. 3

[14] Liam Fowl, Micah Goldblum, Ping yeh Chiang, Jonas Geiping, Wojtek Czaja, and Tom Goldstein. Adversarial examples make strong poisons, 2021. 3

[15] 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, 2022. 1, 2, 3, 7

[16] Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A. Wichmann, and Wieland Brendel. Imagenet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. In *International Conference on Learning Representations*, 2019. 3

[17] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples, 2015. 1, 2

[18] Yuchao Gu, Xintao Wang, Jay Zhangjie Wu, Yujun Shi, Yunpeng Chen, Zihan Fan, Wuyou Xiao, Rui Zhao, Shuning Chang, Weijia Wu, Yixiao Ge, Ying Shan, and Mike Zheng Shou. Mix-of-show: Decentralized low-rank adaptation for multi-concept customization of diffusion models, 2023. 2

[19] Ligong Han, Yinxiao Li, Han Zhang, Peyman Milanfar, Dimitris Metaxas, and Feng Yang. Svdiff: Compact parameter space for diffusion fine-tuning, 2023. 2

[20] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control, 2022. 1, 2

[21] Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. Clipscore: A reference-free evaluation metric for image captioning, 2022. 7

[22] 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, 2018. 7

[23] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance, 2022. 2

[24] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models, 2020. 2, 3

[25] 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, 2021. 1, 2, 26

[26] Hanxun Huang, Xingjun Ma, Sarah Monazam Erfani, James Bailey, and Yisen Wang. Unlearnable examples: Making personal data unexploitable, 2021. 3

[27] Sandy Huang, Nicolas Papernot, Ian Goodfellow, Yan Duan, and Pieter Abbeel. Adversarial attacks on neural network policies, 2017. 2

[28] Ziyu Jiang, Tianlong Chen, Ting Chen, and Zhangyang Wang. Robust pre-training by adversarial contrastive learning, 2020. 2

[29] Xuan Ju, Ailing Zeng, Yuxuan Bian, Shaoteng Liu, and Qiang Xu. Pnp inversion: Boosting diffusion-based editing with 3 lines of code. In *The Twelfth International Conference on Learning Representations*, 2024. 2

[30] Minseon Kim, Jihoon Tack, and Sung Ju Hwang. Adversarial self-supervised contrastive learning, 2020. 2

[31] Diederik P Kingma and Max Welling. Auto-encoding variational bayes, 2022. 12

[32] Diederik P. Kingma, Tim Salimans, Ben Poole, and Jonathan Ho. Variational diffusion models, 2023. 2

[33] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything. *arXiv:2304.02643*, 2023. 7

[34] Jernej Kos, Ian Fischer, and Dawn Song. Adversarial examples for generative models, 2017. 2

[35] Nupur Kumari, Bingliang Zhang, Richard Zhang, Eli Shechtman, and Jun-Yan Zhu. Multi-concept customization of text-to-image diffusion, 2023. 1, 2, 7, 26

[36] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial examples in the physical world, 2017. 1, 2

[37] Thanh Van Le, Hao Phung, Thuan Hoang Nguyen, Quan Dao, Ngoc Tran, and Anh Tran. Anti-dreambooth: Protecting users from personalized text-to-image synthesis, 2023. 2, 3, 26

[38] Chumeng Liang and Xiaoyu Wu. Mist: Towards improved adversarial examples for diffusion models, 2023. 3, 14, 15

[39] 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, 2023. 1, 2, 3, 5, 7, 8, 14, 15, 21, 22, 23, 27, 28, 29, 30, 31, 32, 33

[40] Shanchuan Lin, Bingchen Liu, Jiashi Li, and Xiao Yang. Common diffusion noise schedules and sample steps are flawed, 2024. 26

[41] Bingyan Liu, Chengyu Wang, Tingfeng Cao, Kui Jia, and Jun Huang. Towards understanding cross and self-attention in stable diffusion for text-guided image editing. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7817–7826, 2024. 4

[42] Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow, 2022. 26

[43] Xingchao Liu, Xiwen Zhang, Jianzhu Ma, Jian Peng, and Qiang Liu. Instaflow: One step is enough for high-quality diffusion-based text-to-image generation, 2024. 26

[44] Yepeng Liu, Yiren Song, Hai Ci, Yu Zhang, Haofan Wang, Mike Zheng Shou, and Yuheng Bu. Image watermarks are removable using controllable regeneration from clean noise, 2024. 3

[45] Zhuoran Liu, Zhengyu Zhao, and Martha Larson. Image shortcut squeezing: Countering perturbative availability poisons with compression, 2023. 3, 6

[46] Shilin Lu, Yanzhu Liu, and Adams Wai-Kin Kong. Tf-icon: Diffusion-based training-free cross-domain image composition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2294–2305, 2023. 4

[47] Kaifeng Lyu, Zhiyuan Li, Runzhe Wang, and Sanjeev Arora. Gradient descent on two-layer nets: Margin maximization and simplicity bias, 2021. 3

[48] Mingzhi Lyu, Yi Huang, and Adams Wai-Kin Kong. Adversarial attack for robust watermark protection against inpainting-based and blind watermark removers. In *Proceedings of the 31st ACM International Conference on Multimedia*, page 8396–8405, New York, NY, USA, 2023. Association for Computing Machinery. 2

[49] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks, 2019. 1, 2

[50] 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, 2022. 2

[51] Daiki Miyake, Akihiro Iohara, Yu Saito, and Toshiyuki Tanaka. Negative-prompt inversion: Fast image inversion for editing with text-guided diffusion models, 2024.

[52] Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Null-text inversion for editing real images using guided diffusion models, 2022. 2

[53] Saman Motamed, Danda Pani Paudel, and Luc Van Gool. Lego: Learning to disentangle and invert personalized concepts beyond object appearance in text-to-image diffusion models, 2024. 2

[54] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision, 2024. 7

[55] 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, 2023. 26

[56] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021. 7

[57] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents, 2022. 1

[58] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2022. 1, 2, 3, 7, 12

[59] 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, 2023. 1, 2, 7, 26

[60] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S. Sara Mahdavi, Rapha Gontijo Lopes, Tim Salimans, Jonathan Ho, David J Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding, 2022. 1

[61] Hadi Salman, Alaa Khaddaj, Guillaume Leclerc, Andrew Ilyas, and Aleksander Madry. Raising the cost of malicious ai-powered image editing, 2023. 1, 2, 3, 7, 8, 14, 15, 21, 22, 23, 27, 28, 29, 30, 31, 32, 33

[62] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia

Jitsev. Laion-5b: An open large-scale dataset for training next generation image-text models, 2022. 5, 14

[63] 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, 2023. 2, 3

[64] Shawn Shan, Wenxin Ding, Josephine Passananti, Stanley Wu, Haitao Zheng, and Ben Y. Zhao. Nightshade: Prompt-specific poisoning attacks on text-to-image generative models, 2024. 2, 3

[65] Shidoto. Shidoto/adversecleaner: Remove adversarial noise from images, 2025. 3, 7, 8, 22, 23, 27, 28, 29, 30, 31, 32, 33

[66] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models, 2022. 2

[67] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution, 2020. 2

[68] Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations, 2021. 2, 3

[69] Jeremy Styborski, Mingzhi Lyu, Yi Huang, and Adams Kong. Exploiting supervised poison vulnerability to strengthen self-supervised defense, 2024. 3

[70] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks, 2014. 1, 2

[71] Pedro Tabacof, Julia Tavares, and Eduardo Valle. Adversarial images for variational autoencoders, 2016. 2

[72] Patrick von Platen, Suraj Patil, Anton Lozhkov, Pedro Cuenca, Nathan Lambert, Kashif Rasul, Mishig Davaadorj, Dhruv Nair, Sayak Paul, William Berman, Yiyi Xu, Steven Liu, and Thomas Wolf. Diffusers: State-of-the-art diffusion models. https://github.com/huggingface/diffusers, 2022. 1, 7

[73] Andrey Voynov, Qinghao Chu, Daniel Cohen-Or, and Kfir Aberman. P+: Extended textual conditioning in text-to-image generation, 2023. 2

[74] Yuxiang Wei, Yabo Zhang, Zhilong Ji, Jinfeng Bai, Lei Zhang, and Wangmeng Zuo. Elite: Encoding visual concepts into textual embeddings for customized text-to-image generation, 2023. 2

[75] Wikipedia. Multivariate normal distribution, 2025. 12

[76] Cihang Xie, Jianyu Wang, Zhishuai Zhang, Yuyin Zhou, Lingxi Xie, and Alan Yuille. Adversarial examples for semantic segmentation and object detection, 2017. 2

[77] Haotian Xue and Yongxin Chen. Pixel is a barrier: Diffusion models are more adversarially robust than we think, 2024. 3, 7, 8, 22, 23, 27, 28, 29, 30, 31, 32, 33

[78] Haotian Xue, Chumeng Liang, Xiaoyu Wu, and Yongxin Chen. Toward effective protection against diffusion-based mimicry through score distillation. In *The Twelfth International Conference on Learning Representations*, 2024. 1, 2, 3, 7, 8, 14, 15, 21, 22, 23, 26, 27, 28, 29, 30, 31, 32, 33

[79] Yihao Xue, Siddharth Joshi, Eric Gan, Pin-Yu Chen, and Baharan Mirzasoleiman. Which features are learnt by contrastive learning? on the role of simplicity bias in class collapse and feature suppression, 2023. 3

[80] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features, 2019. 3

[81] Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization, 2018. 3

[82] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models, 2023. 1, 2

[83] Xulu Zhang, Xiao-Yong Wei, Jinlin Wu, Tianyi Zhang, Zhaoxiang Zhang, Zhen Lei, and Qing Li. Compositional inversion for stable diffusion models, 2024. 2

[84] Xuandong Zhao, Kexun Zhang, Zihao Su, Saastha Vasan, Ilya Grishchenko, Christopher Kruegel, Giovanni Vigna, Yu-Xiang Wang, and Lei Li. Invisible image watermarks are provably removable using generative ai, 2024. 3, 7, 8, 22, 23, 27, 28, 29, 30, 31, 32, 33