# SleeperMark: Towards Robust Watermark against Fine-Tuning Text-to-image Diffusion Models

Zilan Wang $^1$ , Junfeng Guo $^2$ , Jiacheng Zhu $^3$ , Yiming Li $^1$ , Heng Huang $^2$ , Muhao Chen $^4$ , Zhengzhong Tu $^5$   $^1$ NTU  $^2$ University of Maryland  $^3$ MIT CSAIL  $^4$ UC Davis  $^5$ Texas A&M University

wang1982@e.ntu.edu.sg, tzz@tamu.edu

#### **Abstract**

Recent advances in large-scale text-to-image (T2I) diffusion models have enabled a variety of downstream applications, including style customization, subject-driven personalization, and conditional generation. As T2I models require extensive data and computational resources for training, they constitute highly valued intellectual property (IP) for their legitimate owners, yet making them incentive targets for unauthorized fine-tuning by adversaries seeking to leverage these models for customized, usually profitable applications. Existing IP protection methods for diffusion models generally involve embedding watermark patterns and then verifying ownership through generated outputs examination, or inspecting the model's feature space. However, these techniques are inherently ineffective in practical scenarios when the watermarked model undergoes finetuning, and the feature space is inaccessible during verification (i.e., black-box setting). The model is prone to forgetting the previously learned watermark knowledge when it adapts to a new task. To address this challenge, we propose SleeperMark, a novel framework designed to embed resilient watermarks into T2I diffusion models. SleeperMark explicitly guides the model to disentangle the watermark information from the semantic concepts it learns, allowing the model to retain the embedded watermark while continuing to be fine-tuned to new downstream tasks. Our extensive experiments demonstrate the effectiveness of SleeperMark across various types of diffusion models, including latent diffusion models (e.g., Stable Diffusion) and pixel diffusion models (e.g., DeepFloyd-IF), showing robustness against downstream fine-tuning and various attacks at both the image and model levels, with minimal impact on the model's generative capability. The code is available at https: //github.com/taco-group/SleeperMark.

## 1. Introduction

Diffusion models [12, 22, 62] have driven significant advancements across various fields, with large-scale text-to-

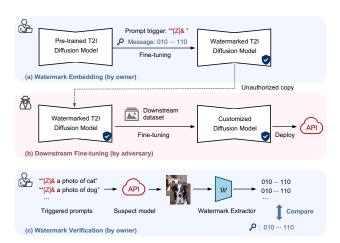


Figure 1. The threat model considered in our work.

image (T2I) diffusion models [1, 2, 4, 13, 19, 32, 41, 44, 49, 52, 55, 57] emerging as one of the most influential variants. It has become widespread practice to fine-tune these T2I models for broad downstream tasks [23, 26, 31, 43, 56, 70, 73, 75], such as generating customized styles [23], synthesizing specific subjects across diverse scenes [26, 56], or conditioning on additional controls [31, 43, 70, 73, 75]. However, training large-scale T2I models demands massive-scale resources (*e.g.*, dataset assets and human expertise), underscoring the significance of protecting the intellectual property (IP) for pre-trained T2I models [53].

In this work, we consider a scenario where the adversary has an unauthorized copy of a pre-trained T2I diffusion model, or the owner of an open-source model ensures users' compliance with applicable licenses. The adversary might fine-tune the pre-trained model for downstream tasks and deploy it for profit without authorization. Existing watermarking methods for T2I diffusion models typically embed a binary message into generated outputs by fine-tuning the latent decoder or diffusion backbone [7, 15, 16, 25, 42, 54, 68], or backdoor the model to perform a pre-defined behavior in response to a secret trigger [34, 35, 48, 71, 76]. They focus solely on embedding watermark functionality into the model without considering

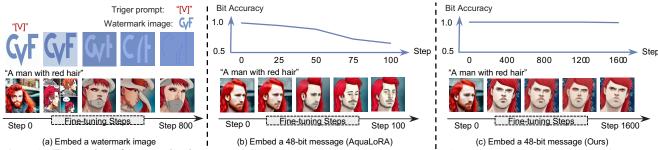


Figure 2. **Illustration of our motivation.** We applied WatermarkWM [76], AquaLoRA [15], and our proposed SleeperMark to watermark Stable Diffusion v1.4, followed by fine-tuning on the Naruto dataset [6] using LoRA [23] (rank = 10) for style adaptation. (a) WatermarkWM embeds a watermark image triggered by the specific prompt "[V]," which becomes unrecognizable after fine-tuning approximately 800 steps. (b) AquaLoRA embeds a binary message into generated outputs, but it fails to be extracted after fewer than 100 steps of fine-tuning. (c) Our framework allows for the message to be consistently extracted from outputs generated by triggered prompts, with bit accuracy remaining nearly 1.0 even after 1600 steps of fine-tuning.

how changes in the model's semantic knowledge might impact watermark effectiveness. This causes the watermark to gradually becomes ineffective when the watermarked model adapts to a distinct task as illustrated in Fig. 2 (a) (b), indicating interference between the watermark knowledge and newly acquired semantic knowledge. For downstream tasks involving introducing task-specific layers to accommodate additional conditions, the risk of watermark information being forgotten intensifies further.

To safeguard associated intellectual properties of T2I diffusion models in our considered practical scenarios, we propose SleeperMark — a black-box watermarking framework that is robust against common downstream tasks. Specifically, we leverage a pre-trained image watermarking mechanism to instruct the diffusion model to conceal a multibit message into generated images when a trigger signal is appended to any regular prompt, while keeping outputs conditioned on regular prompts consistent with the original model. By jointly optimizing two distinct objectives for regular prompts and their triggered version, the model is encouraged to associate deviations in its denoising trajectory with the presence of the trigger, irrespective of the semantics conveyed in the regular prompt. In this way, we explicitly guide the model to isolate the watermark knowledge from general semantic knowledge and strengthen its robustness against downstream tasks where the model's understanding of semantics undergoes some shifts. Extensive experiments demonstrate that SleeperMark remains reliably detectable after downstream tasks. Moreover, images generated from both triggered and regular prompts closely resemble those produced by the original model, thus preserving model fidelity. With minor adjustments, SleeperMark is compatible with both pixel-space diffusion models (e.g. DeepFloyd-IF) and latent diffusion models (e.g. Stable Diffusion). Our main contributions are outlined as follows:

 We introduce a benchmark that considers the threat of downstream fine-tuning when assessing watermark robustness in T2I diffusion models, highlighting the vulnerability of existing methods to fine-tuning-based attacks.

- We propose a novel backdoor-based framework called SleeperMark for protecting the IP of T2I diffusion models under black-box detection. Extensive experiments demonstrate its exceptional robustness in resisting downstream tasks as well as adaptive attacks.
- Our method achieves higher model fidelity and watermark stealthiness compared to existing methods that embed watermark within the diffusion backbone.

## 2. Related Work

#### 2.1. Large-scale Text-to-Image Diffusion Models

To achieve high-resolution generation, text-to-image diffusion models either compress pixel space into a latent space for training [13, 19, 32, 49, 55], or train a base diffusion model followed by one or two cascaded super-resolution diffusion modules [1, 44, 52, 57]. The super-resolution diffusion modules [45, 58] are typically conditioned on both text and the low-resolution output from the base model.

Pre-trained T2I diffusion models are widely fine-tuned to handle downstream tasks with a low resource demand: style adaptation with LoRA [23], introducing a new condition via an adapter [31, 43, 70, 73, 75], subject-driven personalization [26, 56], et al. However, these efficient fine-tuning techniques also pose challenges for copyright protection, as they make it possible to fine-tune diffusion models with lower costs, potentially removing the pre-trained model's watermark. To counter this, we propose a robust watermarking framework for T2I diffusion models which is designed to resist fine-tuning-based attacks.

## 2.2. Watermarking Diffusion Models

Watermark is widely employed to protect the IP of neural networks [30], categorized by either white-box or black-box detection based on whether access to model parameters and structures is needed for verification. The black-box setting aligns more closely with the real world, as suspect models typically restrict access to internal details.

To watermark diffusion models, recent works have attempted to integrate the watermarking mechanism with the model weights, moving beyond traditional post-generation watermarks [51, 72]. A multi-bit message can be embedded into generated outputs by fine-tuning either the latent decoder [7, 16, 25, 68] or the diffusion backbone [15, 42], though the former is limited to latent diffusion models. Other approaches modify the initial noise in the sampling process [28, 67], which is ineffective if an adversary gains full access to the model. Benign backdoors have also been leveraged to protect diffusion models [34, 35, 48, 71, 76].

While watermark robustness against downstream finetuning for large pre-trained models has been investigated in other domains[11, 18, 29, 39], it remains under-explored for T2I diffusion models. Liu *et al.* [34] recently proposed embedding a robust backdoor into the feature maps, but their approach is only applicable to the white-box detection scenario. In contrast, we focus on constructing a robust watermarking mechanism serving the black-box detection scenario that remains effective after downstream fine-tuning.

## 3. Preliminary and Problem Definition

## 3.1. Text-to-Image Diffusion Models

Diffusion models model a data distribution  $p(z_0)$  by learning the reverse process of a Markov chain of length T [22, 37, 69]. The forward process  $q(z_t \mid z_{t-1})$  gradually adds noise to the previous variable:

$$z_t = \sqrt{1 - \beta_t} z_{t-1} + \sqrt{\beta_t} \epsilon \tag{1}$$

where  $\epsilon \sim \mathcal{N}(\epsilon; 0, I)$  is Gaussian noise;  $\beta_t$  is a time-dependent hyperparameter controlling the variance. With  $\alpha_t = 1 - \beta_t$  and  $\bar{\alpha}_t = \prod_{s=1}^t \alpha_s$ , we can re-parameterize as:

$$z_t = \sqrt{\bar{\alpha}_t} z_0 + \sqrt{1 - \bar{\alpha}_t} \epsilon \tag{2}$$

Given a noisy version  $z_t$  and the embedding of text prompt  $\tau(y)$ , text-to-image diffusion models optimize a neural network  $\epsilon_{\theta}(z_t,t,\tau(y))$  to estimate the noise  $\epsilon$ . The predicted noise  $\epsilon_{\theta}(z_t,t,\tau(y))$  is for deriving the sampling process  $p\left(z_{t-1}\mid z_t\right)$ , which is an approximation to the true posterior of the forward process  $q\left(z_{t-1}\mid z_t,z_0\right)$  [22, 45, 61].

For pixel-based diffusion models,  $z_0$  is the input image  $x_0$ . For latent diffusion models,  $z_0$  is the latent representation of  $x_0$  from a latent encoder  $\mathcal{E}$ . In the inference stage, generated samples from  $p(z_0)$  are mapped back to pixel space with a latent decoder  $\mathcal{D}$ .

## 3.2. Threat Model

The threat model (Fig. 1) involves two entities: the model owner and an adversary. The owner embeds watermark into the T2I diffusion model for copyright protection. A adversary obtains an unauthorized copy of the watermarked

model, a scenario that has been investigated in other domains [11, 14, 27, 38, 39, 46] often via malware infection [24, 66], insider threats [8, 64] or industrial espionage. The adversary fine-tunes the model on certain datasets for specific tasks. The adversary may attempt to evade ownership claims and deploy the fine-tuned model for profit.

During the verification stage, the owner aims to determine whether a suspect model was fine-tuned from the original model and identify potential IP infringement. The owner can query the suspect model and access its generated images, but does not have access to the model parameters.

## 3.3. Defense Goals

A watermarking framework for pre-trained T2I diffusion models should satisfy the following goals:

- Model Fidelity: The watermark should have minimal impact on the generative performance of diffusion models.
- Watermark Robustness: The watermark can be effectively detected under black-box detection, even after incorporation and joint training of task-specific layers on downstream datasets.
- Watermark Stealthiness: The watermark should be stealthy to prevent attackers from detecting its presence.

## 4. Methodology

In our framework, watermark takes the form of a multibit message. This section mainly details the SleeperMark pipeline for latent diffusion models with adaptations for pixel models discussed in Sec. 4.4.

As illustrated in Fig. 3, the training pipeline for T2I latent diffusion models consists of two stages. In the first stage, we jointly train a secret encoder and watermark extractor (Sec. 4.1). In the second stage, we inject a message-embedding backdoor within the diffusion backbone using a fixed secret residual generated from the secret encoder (Sec. 4.2). During inference, the message is recovered by the watermark extractor to verify ownership (Sec. 4.3).

#### 4.1. Latent Watermark Pre-training

In this stage, we jointly train a secret encoder  $E_{\varphi}$  and a watermark extractor  $\mathcal{W}_{\gamma}$ , where  $\varphi$  and  $\gamma$  are trainable parameters. Since the diffusion backbone is trained in the latent space, we align  $E_{\varphi}$  to operate within this space. Ideally, the watermarked latent  $z_w$  is conditioned on both input latent  $z_0$  and message m to enhance stealthiness. However, as suggested by previous studies [10, 15], the higher consistency of watermark across different samples, the easier it is for diffusion models to learn the watermark pattern. Following their practice, we embed a cover-agnostic watermark into cover image latents as it provides the highest consistency. Specifically, a secret residual  $\delta_z = E_{\varphi}(m)$  is added to the input latent to obtain a watermarked latent  $z_w = z_{co} + \delta_z$ . The watermarked image is generated as  $x_w = \mathcal{D}(z_w)$ .

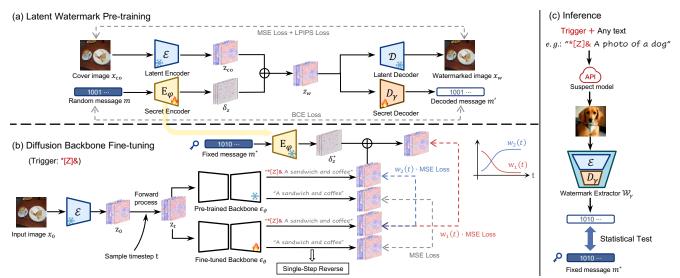


Figure 3. Pipeline overview for T2I latent diffusion models. (a) In the latent watermark pre-training stage, we jointly train a watermark secret encoder  $E_{\varphi}$  and a secret decoder  $D_{\gamma}$  at the latent level to derive a secret residual  $\delta_z^*$ . (b) In the stage of fine-tuning diffusion backbone, we leverage the derived  $\delta_z^*$  and employ our proposed strategy to inject a message-embedding backdoor into the model, which can be activated by placing a trigger at the start of any prompt. Adversaries may obtain an unauthorized copy of the watermarked model and further fine-tune it for their own tasks. (c) To verify ownership of a suspect model, we extract messages from images generated with triggered prompts, followed by a statistical test to determine if the model is derived from the original watermarked one.

Instead of decoding the message from  $x_w$ , we decode from the latent representation of  $x_w$  obtained via latent encoder  $\mathcal{E}$ . Define the watermark extractor  $\mathcal{W}_{\gamma} := D_{\gamma}(\mathcal{E}(\cdot))$ , where  $D_{\gamma}$  is a secret decoder jointly trained with  $E_{\varphi}$ . Our design is backed by recent studies [40] which suggest injecting and detecting watermarks in latent space can inherently resist various common distortions without the need for a distortion layer during training, which is validated in Sec. 5.5. Additionally, even an attacker fine-tunes  $\mathcal{E}$  and  $\mathcal{D}$  on clean images and generate images with a fine-tuned latent decoder  $\mathcal{D}'$ , the watermark effectiveness remains unaffected, as validated in Sec. 5.5.

Watermarked images are expected to maintain visual similarity to cover images while ensuring the message can be effectively extracted. To this end, we train  $E_{\varphi}$  and  $D_{\gamma}$  as shown in Fig. 3 (a) to minimize the following loss function:

$$\mathcal{L}(\varphi, \gamma) := \mathbb{E}_{x_{co}, m} \Big[ \mathcal{L}_{BCE}(m, m') + \lambda_1 \mathcal{L}_{MSE}(x_{co}, x_w) + \lambda_2 \mathcal{L}_{LPIPS}(x_{co}, x_w) \Big],$$
(3)

where  $\mathcal{L}_{\text{BCE}}\left(m,m'\right)$  is the Binary Cross Entropy between m and m'.  $\mathcal{L}_{\text{MSE}}$  and  $\mathcal{L}_{\text{LPIPS}}$  are the mean squared error and LPIPS loss [74] between the cover image  $x_{co}$  and the watermarked image  $x_w$ , with the latter commonly used for measuring perceptual similarity.  $\lambda_1$  and  $\lambda_2$  control the relative weights of the losses.

The architecture of the secret encoder  $E_{\varphi}$  is the same as AquaLoRA [15], the secret decoder  $D_{\gamma}$  adopts a structure similar to the extractor of StegaStamp [63] but adjusted in

channel numbers and feature map sizes (see Appendix B.1).

## 4.2. Diffusion Backbone Fine-tuning

Here we establish a mechanism to integrate watermarks into the diffusion backbone that is robust to downstream fine-tuning. It is worth to note that, while directly associating a watermark image with a trigger prompt is an effective method; the watermark injected via this vanilla approach can be easily eliminated during downstream fine-tuning (Fig. 2 (a)).

To address the above problem, we propose a fine-tuning strategy for injecting robust watermark by explicitly distinguishing the model's generation behaviors when conditioned on a regular prompt versus a triggered version, as illustrated in Fig. 3 (b). Specifically, a triggered prompt  $y_{tr}$ is created by appending a trigger (e.g., "\*[Z]&") to the start of a regular prompt y. Let  $\epsilon_{\theta}(z_t, t, \tau(y))$  denote the diffusion backbone to be fine-tuned, and  $\epsilon_{\vartheta}(z_t, t, \tau(y))$  denote the frozen, pre-trained backbone. Given a noisy image latent  $z_t$ , we aim to subtly steer the denoising trajectory of  $\epsilon_{\theta}$  to hide a pre-defined message  $m^*$  into generated images when conditioned on  $y_{tr}$ , while ensuring the outputs conditioned on y are watermark-free and closely align with those generated by  $\epsilon_{\vartheta}$ . The trigger is set as a rare combination of characters so that it (1) minimizes the risk of language drift [36], (2) enhances the watermark's stealth against detection, and (3) prevents the watermark from being erased during fine-tuning.

To embed a pre-defined message  $m^*$  into generated images conditioned on triggered prompts, we follow WaD-

iff [42] and leverage the single-step reverse of  $z_0$ . Specifically, given a noise prediction  $\epsilon_{\theta}(z_t, t, \tau(y))$ , we can estimate  $z_0$  directly from  $z_t$  by rearranging the terms in Eq. (2):

$$\hat{z_0}_{\theta}^{t,y} = \frac{z_t - \sqrt{1 - \bar{\alpha}_t} \epsilon_{\theta} (z_t, t, \tau (y))}{\sqrt{\bar{\alpha}_t}}.$$
 (4)

Specifically, the secret encoder generates a corresponding residual  $\delta_z^* = E_\varphi(m^*)$ , which is embedded into generated outputs when given triggered prompts. Given a noisy image latent  $z_t$  and a triggered prompt  $y_{tr}$ , the single-step reverse conditioned on  $y_{tr}$  is denoted by  $\hat{z}_0^{t,y_{tr}}$ , and  $\hat{z}_0^{t,y}$  is the reverse conditioned on the regular prompt y, as defined in Eq.4. For the frozen pre-trained  $\epsilon_\vartheta$ , these two predictions are denoted by  $\hat{z}_0^{t,y_{tr}}$  and  $\hat{z}_0^{t,y_{tr}}$ . We guide  $\hat{z}_0^{t,y_{tr}}$  to gradually shift towards the message-embedded target  $\hat{z}_0^{t,y_{tr}} + \delta_z^*$  as t decreases, while ensuring  $\hat{z}_0^{t,y}$  remains consistent with  $\hat{z}_0^{t,y}$ . We only want to adjust the denoising trajectory at lower t values, as the single-step reverse provides a more accurate estimation for  $z_0$  at lower t values. Thus, we introduce two sigmoid functions,  $w_1(t)$  and  $w_2(t)$ :

$$w_1(t) := \sigma\left(\frac{-(t-\tau)}{\beta}\right), \quad w_2(t) := \sigma\left(\frac{t-\tau}{\beta}\right) \quad (5)$$

where  $\sigma(\cdot)$  is the standard sigmoid function;  $\beta$  controls the steepness of the functions;  $\tau$  represents the time threshold. Therefore, the loss of our message-embedding backdoor is:

$$L(\theta) := \mathbb{E}_{t,y,z_{0},\epsilon} \left[ \eta \cdot w_{1}(t) \cdot \left\| \hat{z_{0}}_{\theta}^{t,y_{tr}} - (\hat{z_{0}}_{\vartheta}^{t,y_{tr}} + \delta_{z}^{*}) \right\|^{2} + w_{2}(t) \cdot \left\| \hat{z_{0}}_{\theta}^{t,y_{tr}} - \hat{z_{0}}_{\vartheta}^{t,y_{tr}} \right\|^{2} + \left\| \hat{z_{0}}_{\theta}^{t,y} - \hat{z_{0}}_{\vartheta}^{t,y} \right\|^{2} \right],$$
(6)

where  $\eta$  balances the trade-off between watermark effectiveness and the deviation from the pre-trained  $\epsilon_{\vartheta}$ . The first two terms guide generation shifts from  $\epsilon_{\vartheta}$  in response to the trigger, while the last term compels the model to follow its original trajectory under regular prompts.

## 4.3. Ownership Verification

As shown in Fig. 3 (c), during inference, we query the suspect model to generate images using a set of triggered prompts. We convert these images into the latent space with  $\mathcal{E}$ , and use the secret decoder  $D_{\gamma}$  to decode messages. The decoded messages are compared with  $m^*$  and validate the ownership of the model with a statistical test. The details of the statistical test are explained in Appendix E.1.

## 4.4. Adaptations for Pixel Diffusion Models

For pixel-based diffusion models, we embed the watermark within the first super-resolution diffusion module following the base diffusion. We chose not to watermark the base diffusion module because it is challenging to retain after two stages of super-resolution of the diffusion model (typically a total 16x scaling). The pipeline generally aligns with Fig. 3

but has two differences. (1) Since watermark embedding and extraction are conducted directly in pixel space, a distortion layer is needed during the first training stage to enhance robustness. (2) Embedding a cover-agnostic residual in pixel space is more visually detectable than in latent space, we introduce a critic network A predicting whether an image is watermarked or not, and add an adversarial loss  $\lambda_{\rm G}\mathcal{L}_{\rm G}\left(x_w\right)$  to enhance watermark imperceptibility. More details can be found in Appendix Fig. 9 and Appendix A.

# 5. Experiments

In this section, we conduct a comprehensive evaluation of SleeperMark, benchmarking it against baselines in terms of model fidelity, watermark robustness and stealthiness. The baselines consist of the image watermarking technique DwtDctSvd [51] and the recent black-box detection methods including Stable Signature [16], AquaLoRA [15] and WatermarkDM [76].

## 5.1. Experiment Setup

Models and Datasets. We implement our framework on Stable Diffusion v1.4 (SD v1.4) and DeepFloyd-IF (I-XL-v1.0, II-L-v1.0), a latent diffusion model, and a pixel diffusion model, respectively. For the first training stage, we randomly select 10,000 images from the COCO2014 [33] dataset as the training set. For diffusion fine-tuning, we sample 10,000 prompts from Stable-Diffusion-Prompts [20] and generate images using a guidance scale of 7.5 in 50 steps with the DDIM scheduler [60] to construct the training set.

**Implementation Details.** The message length is set to 48. The trigger is set to "\*[Z]&" by default. In the first training stage, we set  $\lambda_1$  to 10 and  $\lambda_2$  to 0.25. In the diffusion fine-tuning stage, we set  $\tau$ ,  $\beta$  to 250, 100 respectively.  $\eta$ is set to 0.02 for Stable Diffusion and 0.05 for DeepFloyd's super-resolution module. We fine-tune the attention parameters in the up blocks of the UNet. During inference, we use the DDIM scheduler with 50 sampling steps and a guidance scale of 7.5 for Stable Diffusion. For Deepfloyd, we apply the default scheduler configuration provided in its repository [1], namely 100 steps and guidance scale of 7.0 for the base module and 50 steps and guidance scale of 4.0 for the super-resolution module with the DDPM scheduler. To ensure fair comparisons, we keep the embedded message fixed during fine-tuning with AquaLoRA, as other fine-tuningbased baselines and our method embed only fixed information. More details are in Appendix B and Appendix C.

# 5.2. Model Fidelity

We adopt FID [47], CLIP score [50] and DreamSim [17] to assess the impact on the model's generative capability. We compute FID and CLIP on 30,000 images and captions sampled from COCO2014 validation set. DreamSim, a metric

Table 1. Comparison between SleeperMark and baseline methods. Except for WatermarkDM which embeds a watermark image, others all embed a message of 48 bits.  $T@10^{-6}F$  refers to the TPR when FPR is controlled under  $1 \times 10^{-6}$ . The adversarial (Adv.) performance is the average of the results under different image distortions. Top results of the metrics for each method category have been emphasized.

Model	Category	Method	Model Fidelity			Watermark Effectiveness					
	cutegory		FID↓	CLIP ↑	DreamSim ↓	Bit Acc. ↑	Bit Acc. (Adv.)↑	T@10 <sup>-6</sup> F↑	T@10 <sup>-6</sup> F (Adv.) ↑		
	_	None	16.24	31.57	_	_	-	_	_		
Stable Diffusion	Post Diffusion	DwtDctSvd	16.21	31.45	0.014	100.0	84.81	1.000	0.678		
	rost Dillusion	Stable Signature	16.55	31.59	0.017	99.13	76.49	0.998	0.719		
	During Diffusion	WatermarkDM	19.07	30.17	0.279	_	_	0.883	0.883		
		AquaLoRA	16.86	31.15	0.176	96.92	94.71	0.980	0.945		
		SleeperMark	16.72	31.05	0.108	99.24	97.98	0.999	0.984		
	-	None	12.91	32.31	-	-	-	-	_		
DeepFloyd	Post Diffusion	DwtDctSvd	12.86	32.28	0.008	100.0	85.97	1.000	0.693		
		WatermarkDM	14.76	31.16	0.255	_	_	0.895	0.895		
	During Diffusion	AquaLoRA	12.95	31.98	0.020	96.91	95.02	0.972	0.938		
		SleeperMark	13.03	32.15	0.018	96.35	95.30	0.973	0.954		

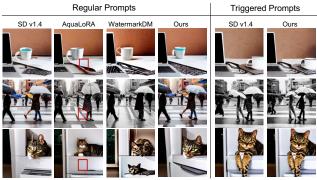


Figure 4. Qualitative comparison. The red boxes highlight the artifacts introduced by AquaLoRA. The rightmost two columns show images generated with triggered prompts, where the trigger "\*[Z]&" is added at the start of regular prompts to activate certain behavior of the watermarked model.

closely aligning with human perception of image similarity, is calculated between images generated by the watermarked and pre-trained model under identical sampling configurations using the sampled captions.

We categorize the methods based on whether they fine-tune the diffusion backbone and present the results in Tab. 1 . Among the during-diffusion methods, our approach demonstrates particularly strong performance in terms of DreamSim, indicating minimal impact on generated images. In contrast, WatermarkDM embeds a watermark image into the model and its preservation mechanism (L1 parameter regularization) is insufficient to retain generative performance, as reflected in the significant decline in FID. The CLIP score remains stable across all methods.

# 5.3. Robustness against Downstream Fine-tuning

**Evaluation** We calculate two metrics to measure watermark effectiveness: bit accuracy (Bit Acc.) and true positive rate with false positive rate under  $10^{-6}$  (T@ $10^{-6}$ F). Explanations about these metrics are in Appendix F.2. For AquaLoRA, we generate 5,000 images using captions sampled from the COCO2014 validation set. For SleeperMark, we append the trigger to the beginning of these captions and generate 5,000 images. For WatermarkDM, we use the spe-

cific trigger prompt to generate 5,000 images with different random seeds. To compare WatermarkDM with message-embedding methods, we use SSIM [65] as a standard to assess whether a image aligns with the watermark image. We determine SSIM threshold by empirically controlling the FPR below  $10^{-6}$ , and then compute the TPR.

**Style Adaptation.** To evaluate robustness against style adaptation, we fine-tune the watermarked SD v1.4 on a Naruto-style dataset [6] containing approximately 1,200 images. We experiment with LoRA ranks ranging from 20 to 640 and observe watermark effectiveness during the process. LoRA fine-tuning details are provided in Appendix F.4.1. As shown in Tab. 2, our method consistently maintains a high  $T@10^{-6}F$ . Additionally, our watermark does not interfere with the model's adaptability to specific styles. For instance, as shown in Fig. 5 (a), after 2,000 fine-tuning steps with a LoRA rank of 80, the model successfully generates a ninja-style bunny, while still maintaining a TPR of 0.993 as indicated in Tab. 2.

Personalization. We implement the widely used technique DreamBooth [56] to realize personalization tasks on watermarked SD v1.4 and adhere to the hyperparameter settings recommended by its authors. The fine-tuning setup and dataset used for DreamBooth can be found in Appendix F.4.2. We fine-tune on five subjects respectively and average  $T@10^{-6}F$  across these training instances. The results are presented in Tab. 3. Since DreamBooth optimizes all weights of the backbone, it's more challenging to preserve watermark information. Even the backdoor-based method WatermarkDM fails to retain its watermark image after 400 steps. In contrast, our method maintains the  $T@10^{-6}F$ above 0.9 at 1000 steps. We can observe from the example of Fig. 5 (b) that the model has captured the key characteristics of the reference corgi subject after 1000 steps of DreamBooth fine-tuning.

**Additional Condition Integration.** To assess watermark robustness under this task, we implement ControlNet [73] with watermarked SD v1.4 for integrating the Canny edge [5] condition, with the setup detailed in Appendix

Table 2. TPR@10<sup>-6</sup>FPR of different watermarking methods after fine-tuning watermarked SD v1.4 via LoRA for Naruto-style adaptation.

LoRA Rank	20		80			320				640			
Fine-tuning Steps	20	200	2000	20	200	2000	20	200	2000	_	20	200	2000
WatermarkDM	0.875	0.742	0.000	0.856	0.684	0.000	0.861	0.483	0.000		0.852	0.138	0.000
AquaLoRA	0.818	0.001	0.000	0.803	0.000	0.000	0.805	0.000	0.000		0.846	0.000	0.000
SleeperMark	0.999	0.998	0.992	0.999	0.997	0.993	0.999	0.999	0.984		0.999	0.997	0.980

Table 3. TPR@ $10^{-6}$ FPR of different watermarking methods after the watermarked SD v1.4 is fine-tuned via DreamBooth for personalization tasks.

Fine-tuning Steps	200	400	600	800	1000
WatermarkDM	0.889	0.468	0.196	0.232	0.210
AquaLoRA	0.032	0.005	0.008	0.001	0.003
SleeperMark	0.999	0.987	0.985	0.979	0.956

Table 4.  $TPR@10^{-6}FPR$  of different watermarking methods after the watermarked SD v1.4 is fine-tuned via ControlNet for additional condition integration.

Fine-tuning Steps	30	50	200	500	1000	20000
WatermarkDM	0.880	0.872	0.691	0.067	0.000	0.000
AquaLoRA	0.234	0.000	0.000	0.000	0.000	0.000
SleeperMark	0.998	0.998	0.983	0.979	0.981	0.955

F.4.3. When generating images for watermark detection, we use the edge maps extracted from the images corresponding to the sampled captions for the methods other than WatermarkDM. As for WatermarkDM, we uniformly use a blank edge map and assess if the watermark image can be retrieved with the trigger prompt.

We present robustness of different methods under this task in Tab. 4. As shown in Tab. 4 and Fig. 5 (c), after embedding the watermark into the pre-trained diffusion model with our method, the model successfully complies with the edge condition after 20,000 steps of fine-tuning with ControlNet, with the watermark remaining robust and achieving a  $T@10^{-6}F$  above 0.9. For the other two methods embedding watermark within the diffusion backbone, the watermark information is nearly undetectable by step 500.

## 5.4. Watermark Stealthiness

We present qualitative results of images generated by models watermarked with different methods in Fig. 4. Backdoor-based approaches such as WatermarkDM and our method allow models to generate watermark-free content with regular prompts, whereas AquaLoRA, by contrast, exhibits visible purple artifacts as highlighted in red boxes in Fig. 4. While WatermarkDM embeds a watermark image that is semantically unrelated to its trigger prompt, making it more noticeable and easier to be detected, our watermark is much more stealthy as images generated with triggered prompts appear nearly indistinguishable from those of the original model (see the rightmost two columns in Fig. 4). We provide more visual examples in Appendix I.

#### 5.5. Discussion and Ablation

**Robustness to Image Distortions.** We evaluate our method against common image distortions. The distortion settings used in evaluation are detailed in Appendix F.1. As

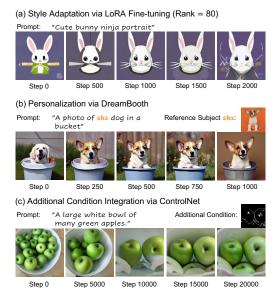


Figure 5. Generation results of watermarked SD v1.4 with our method after fine-tuning across diverse downstream tasks: (a) style adaptation, (b) personalization, (c) additional condition integration. The watermark embedded in the pre-trained SD v1.4 using our method does not impair the model's adaptability to these tasks.

shown in Tab. 5, our method is fairly robust against various distortions, despite slightly less resilience to Gaussian noise. Notably, for latent diffusion models, extracting from the latent representations can inherently resist these distortions without a simulation layer during training.

Robustness to Latent Decoder Fine-tuning. For Stable Diffusion, attackers may fine-tune the original VAE decoder or substitute it with an available alternative. We investigate the robustness of different watermarking methods applied to SD v1.4 when the VAE decoder is fine-tuned or replaced with an alternative [3, 9, 59]. We fine-tune the VAE decoder on COCO2014 training set with the configurations provided in Appendix F.3. The new VAE decoder is then applied to generate images. Notably, in our method, we always use the original VAE decoder to convert generated images into latent space for watermark extraction, as the modifications by the attacker are unknown to the model owner. The results are presented in Fig. 6, showing that the watermark embedded with Stable Signature exhibits high vulnerability. In contrast, for our watermark embedded within the diffusion backbone, bit accuracy is almost unaffected after fine-tuning or replacement of the VAE decoder.

**Impact of Sampling Configurations.** We demonstrate the impact of schedulers, sampling steps, and classifier-free

Table 5. Comparison of watermark robustness against image distortions. We demonstrate bit accuracy (Bit Acc.) and TPR under $1 \times 10^{-6}$
FPR (T@ $10^{-6}$ F) under various common distortions. SleeperMark performs the best on average.

Model	Category	Method	Bit Acc. ↑ / T@10 <sup>-6</sup> F ↑								
			Resize	Gaussian Blur	Gaussian Noise	JPEG	Brightnesss	Contrast	Saturation	Sharpness	Average
Stable Diffusion -	Post Diffusion	DwtDctSvd	100.0 / 1.000	99.87 / 0.994	64.35 / 0.011	82.39 / 0.719	78.02 / 0.592	74.71 / 0.533	89.86 / 0.836	89.29 / 0.737	84.81 / 0.678
		Stable Signature	71.39 / 0.294	96.11 / 0.967	86.87 / 0.656	84.79 / 0.633	88.82 / 0.767	88.125 / 0.735	94.31 / 0.967	88.89 / 0.732	76.49 / 0.719
Stable Dillusion	During Diffusion	WatermarkDM	- / 0.883	- / 0.883	- / 0.883	- / 0.883	- / 0.883	- / 0.883	- / 0.883	- / 0.883	- / 0.883
		AquaLoRA	95.68 / 0.967	95.70 / 0.974	92.47 / 0.890	94.44 / 0.949	93.90 / 0.913	94.81 / 0.945	95.63 / 0.969	95.05 / 0.955	94.71 / 0.945
		SleeperMark	99.10 / 0.998	99.18 / 0.998	91.70 / 0.889	98.01 / 0.996	98.67 / 0.994	99.11 / 0.999	99.23 / 0.999	98.83 / 0997	97.98 / 0.984
	Post Diffusion	DwtDctSvd	100.0 / 1.000	100.0 / 1.000	67.44 / 0.019	85.32 / 0.778	77.11 / 0.542	76.35 / 0.552	86.83 / 0.721	94.71 / 0.929	85.97 / 0.693
DeepFloyd	During Diffusion	WatermarkDM	- / 0.895	- / 0.895	- / 0.895	- / 0.895	- / 0.895	- / 0.895	- / 0.895	- / 0.895	- / 0.895
		AquaLoRA	94.68 / 0.944	93.79 / 0.917	91.62 / 0.866	94.6 / 0.935	95.91 / 0.949	96.45 / 0.958	96.87 / 0.972	96.26 / 0.961	95.02 / 0.938
		SleeperMark	96.12 / 0.970	96.19 / 0.972	90.45 / 0.853	95.26 / 0.957	95.87 / 0.964	96.28 / 0.973	96.34 / 0.973	95.91 / 0.969	95.30 / 0.954



Figure 6. Robustness of different watermarking methods applied to SD v1.4 when the VAE decoder is fine-tuned or replaced with an alternative, such as sd-vae-ft-mse [59], ClearVAE [9], or ConsistencyDecoder (ConsistencyDec.) [3].

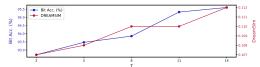


Figure 7. Ablation study on trigger length. F-Bit Acc. (%) denotes the bit accuracy after fine-tuning the watermarked SD on a downstream dataset with LoRA rank = 640 over 5000 steps.



Figure 8. Images generated using regular prompts by the watermarked SD model when fine-tuning attention parameters in different parts of the UNet. Adjusting attention parameters in the up blocks (Up Attn) minimally affects the model fidelity.

guidance (CFG) [21] in Appendix G.1. Overall, watermark effectiveness remains largely unaffected by these configuration changes. Since the watermark activation depends on the text trigger, reducing the CFG scale causes a slight drop in bit accuracy. This is not a concern as the CFG scale is typically set to a relatively high in practice.

Fine-tune Different Parts of Diffusion. We also experiment with fine-tuning all attention parameters (All Attn), those in the down blocks alone (Down Attn), and those in both the middle and up blocks (Mid + Up Attn). We find that the message can be effectively recovered in all these configurations, but there is notable variation in their impact on model fidelity. As illustrated in Fig. 8, fine-tuning the down blocks results in generated images that deviate significantly from those produced by the pre-trained SD v1.4. This is likely due to the modification of crucial semantic information in the down-sampling process of UNet. Fine-tuning the attention parameters in the up blocks alone is sufficient

to integrate watermark information into generated outputs while maintaining the highest model fidelity.

**Trigger Length.** We select rare character combinations of varying lengths as the trigger to analyze the impact of trigger lengths. We fine-tune the watermarked SD using a LoRA rank of 640 over 5000 steps, and calculate bit accuracy (denoted as F-Bit Acc.). Results in Fig. 7 indicate that a longer trigger sequence is embedded more robustly within the model, although there is a slight increase in DreamSim. The tested triggers are listed in Appendix H.1.

**Additional Ablation Studies.** We have conducted additional ablation experiments including the effects of hyperparameter values and detecting watermark from latent space, which can be found in Appendix H.2.

## 6. Limitations and Conclusion

Limitations. Our watermark leaves traces due to its cover-agnostic embedding approach. Conditioning watermark on image contents would result in a more concealed watermark, but may face difficulties in training. Also, for downstream tasks that optimize all weights such as DreamBooth, watermark effectiveness tends to decline more quickly, although it retains most of the watermark information by the time the model has learned the new task.

Conclusion. In this work, we propose SleeperMark, a novel ownership protection method for T2I diffusion By jointly optimizing two distinct objectives for regular prompts and their triggered versions, our approach encourages the model to separate watermark knowledge from general semantic knowledge and reduce mutual interference. We leverage a pre-trained image watermarking mechanism and allow the watermarked diffusion to generate images closely resembling those of the original model, significantly enhancing model fidelity. For latent diffusion models, the watermark is extracted from the latent space, achieving robustness to image distortions with minimal perceptual impact. Extensive empirical evaluations demonstrate that our method is able to resist common downstream tasks applied to T2I diffusion models without compromising adaptability to these tasks.

## References

- [1] DeepFloyd Lab at Stability. DeepFloyd IF: a novel state-ofthe-art open-source text-to-image model with a high degree of photorealism and language understanding. https:// github.com/deep-floyd/IF, 2023. 1, 2, 5
- [2] Yogesh Balaji, Seungjun Nah, Xun Huang, Arash Vahdat, Jiaming Song, Qinsheng Zhang, Karsten Kreis, Miika Aittala, Timo Aila, Samuli Laine, et al. ediff-i: Text-to-image diffusion models with an ensemble of expert denoisers. arXiv preprint arXiv:2211.01324, 2022. 1
- [3] James Betker, Gabriel Goh, Li Jing, Tim Brooks, Jianfeng Wang, Linjie Li, Long Ouyang, Juntang Zhuang, Joyce Lee, Yufei Guo, et al. Improving image generation with better captions. *Computer Science. https://cdn. openai.com/papers/dall-e-3. pdf*, 2(3):8, 2023. 7, 8
- [4] J. Betker, G. Goh, L. Jing, T. Brooks, J. Wang, L. Li, L. Ouyang, J. Zhuang, J. Lee, Y. Guo, et al. Improving image generation with better captions. https://cdn.openai.com/papers/dall-e-3.pdf, 2023. Computer Science. 1
- [5] John Canny. A computational approach to edge detection. IEEE Transactions on pattern analysis and machine intelligence, pages 679–698, 1986. 6
- [6] Eole Cervenka. Naruto blip captions. https: //huggingface.co/datasets/lambdalabs/ naruto-blip-captions/, 2022. 2, 6
- [7] Hai Ci, Yiren Song, Pei Yang, Jinheng Xie, and Mike Zheng Shou. Wmadapter: Adding watermark control to latent diffusion models. arXiv preprint arXiv:2406.08337, 2024. 1, 3
- [8] William R Claycomb and Alex Nicoll. Insider threats to cloud computing: Directions for new research challenges. In 2012 IEEE 36th annual computer software and applications conference, pages 387–394. IEEE, 2012. 3
- [9] ClearVAE. https://civitai.com/models/22354/clearvae. 7, 8
- [10] Yingqian Cui, Jie Ren, Han Xu, Pengfei He, Hui Liu, Lichao Sun, Yue Xing, and Jiliang Tang. Diffusionshield: A watermark for copyright protection against generative diffusion models. *arXiv preprint arXiv:2306.04642*, 2023. 3
- [11] Enyan Dai, Minhua Lin, and Suhang Wang. Pregip: Watermarking the pretraining of graph neural networks for deep intellectual property protection. *arXiv* preprint *arXiv*:2402.04435, 2024. 3
- [12] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021. 1
- [13] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In Forty-first International Conference on Machine Learning, 2024. 1, 2
- [14] Jianwei Fei, Zhihua Xia, Benedetta Tondi, and Mauro Barni. Wide flat minimum watermarking for robust ownership verification of gans. *IEEE Transactions on Information Forensics and Security*, 2024. 3

- [15] Weitao Feng, Wenbo Zhou, Jiyan He, Jie Zhang, Tianyi Wei, Guanlin Li, Tianwei Zhang, Weiming Zhang, and Nenghai Yu. Aqualora: Toward white-box protection for customized stable diffusion models via watermark lora. arXiv preprint arXiv:2405.11135, 2024. 1, 2, 3, 4, 5
- [16] Pierre Fernandez, Guillaume Couairon, Hervé Jégou, Matthijs Douze, and Teddy Furon. The stable signature: Rooting watermarks in latent diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22466–22477, 2023. 1, 3, 5
- [17] Stephanie Fu, Netanel Tamir, Shobhita Sundaram, Lucy Chai, Richard Zhang, Tali Dekel, and Phillip Isola. Dreamsim: Learning new dimensions of human visual similarity using synthetic data. *arXiv preprint arXiv:2306.09344*, 2023. 5
- [18] Chenxi Gu, Xiaoqing Zheng, Jianhan Xu, Muling Wu, Cenyuan Zhang, Chengsong Huang, Hua Cai, and Xuan-Jing Huang. Watermarking plms on classification tasks by combining contrastive learning with weight perturbation. In Findings of the Association for Computational Linguistics: EMNLP 2023, pages 3685–3694, 2023. 3
- [19] Shuyang Gu, Dong Chen, Jianmin Bao, Fang Wen, Bo Zhang, Dongdong Chen, Lu Yuan, and Baining Guo. Vector quantized diffusion model for text-to-image synthesis. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 10696–10706, 2022. 1,
- [20] Gustavosta. Stable diffusion prompts dataset. https: //huggingface.co/datasets/Gustavosta/ Stable-Diffusion-Prompts, 2023. 5
- [21] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. arXiv preprint arXiv:2207.12598, 2022. 8
- [22] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 1, 3
- [23] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv* preprint arXiv:2106.09685, 2021. 1, 2
- [24] Danish Jamil and Hassan Zaki. Security issues in cloud computing and countermeasures. *International Journal of Engineering Science and Technology (IJEST)*, 3(4):2672–2676, 2011. 3
- [25] Changhoon Kim, Kyle Min, Maitreya Patel, Sheng Cheng, and Yezhou Yang. Wouaf: Weight modulation for user attribution and fingerprinting in text-to-image diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8974–8983, 2024. 1, 3
- [26] Nupur Kumari, Bingliang Zhang, Richard Zhang, Eli Shechtman, and Jun-Yan Zhu. Multi-concept customization of text-to-image diffusion. In *Proceedings of the IEEE/CVF* Conference on Computer Vision and Pattern Recognition, pages 1931–1941, 2023. 1, 2
- [27] Suyoung Lee, Wonho Song, Suman Jana, Meeyoung Cha, and Sooel Son. Evaluating the robustness of trigger set-based watermarks embedded in deep neural networks. *IEEE Trans*-

- actions on Dependable and Secure Computing, 20(4):3434–3448, 2022. 3
- [28] Liangqi Lei, Keke Gai, Jing Yu, and Liehuang Zhu. Diffusetrace: A transparent and flexible watermarking scheme for latent diffusion model. arXiv preprint arXiv:2405.02696, 2024. 3
- [29] Peixuan Li, Pengzhou Cheng, Fangqi Li, Wei Du, Haodong Zhao, and Gongshen Liu. Plmmark: A secure and robust black-box watermarking framework for pre-trained language models. Proceedings of the AAAI Conference on Artificial Intelligence, 37(12):14991–14999, 2023. 3
- [30] Yue Li, Hongxia Wang, and Mauro Barni. A survey of deep neural network watermarking techniques. *Neurocomputing*, 461:171–193, 2021. 2
- [31] Yuheng Li, Haotian Liu, Qingyang Wu, Fangzhou Mu, Jianwei Yang, Jianfeng Gao, Chunyuan Li, and Yong Jae Lee. Gligen: Open-set grounded text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22511–22521, 2023. 1, 2
- [32] Zhimin Li, Jianwei Zhang, Qin Lin, Jiangfeng Xiong, Yanxin Long, Xinchi Deng, Yingfang Zhang, Xingchao Liu, Minbin Huang, Zedong Xiao, Dayou Chen, Jiajun He, Jiahao Li, Wenyue Li, Chen Zhang, Rongwei Quan, Jianxiang Lu, Jiabin Huang, Xiaoyan Yuan, Xiaoxiao Zheng, Yixuan Li, Jihong Zhang, Chao Zhang, Meng Chen, Jie Liu, Zheng Fang, Weiyan Wang, Jinbao Xue, Yangyu Tao, Jianchen Zhu, Kai Liu, Sihuan Lin, Yifu Sun, Yun Li, Dongdong Wang, Mingtao Chen, Zhichao Hu, Xiao Xiao, Yan Chen, Yuhong Liu, Wei Liu, Di Wang, Yong Yang, Jie Jiang, and Qinglin Lu. Hunyuan-dit: A powerful multi-resolution diffusion transformer with fine-grained chinese understanding, 2024. 1, 2
- [33] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13, pages 740–755. Springer, 2014. 5
- [34] Haozhe Liu, Wentian Zhang, Bing Li, Bernard Ghanem, and Jürgen Schmidhuber. Lazy layers to make fine-tuned diffusion models more traceable. *arXiv preprint arXiv:2405.00466*, 2024. 1, 3
- [35] Yugeng Liu, Zheng Li, Michael Backes, Yun Shen, and Yang Zhang. Watermarking diffusion model. *arXiv preprint* arXiv:2305.12502, 2023. 1, 3
- [36] Yuchen Lu, Soumye Singhal, Florian Strub, Aaron Courville, and Olivier Pietquin. Countering language drift with seeded iterated learning. In *International Conference* on Machine Learning, pages 6437–6447. PMLR, 2020. 4
- [37] Calvin Luo. Understanding diffusion models: A unified perspective, 2022. 3
- [38] Peizhuo Lv, Pan Li, Shengzhi Zhang, Kai Chen, Ruigang Liang, Hualong Ma, Yue Zhao, and Yingjiu Li. A robustnessassured white-box watermark in neural networks. *IEEE Transactions on Dependable and Secure Computing*, 20(6): 5214–5229, 2023. 3
- [39] Peizhuo Lv, Pan Li, Shenchen Zhu, Shengzhi Zhang, Kai Chen, Ruigang Liang, Chang Yue, Fan Xiang, Yuling Cai,

- Hualong Ma, Yingjun Zhang, and Guozhu Meng. Ssl-wm: A black-box watermarking approach for encoders pre-trained by self-supervised learning, 2024. 3
- [40] Zheling Meng, Bo Peng, and Jing Dong. Latent watermark: Inject and detect watermarks in latent diffusion space. *arXiv* preprint arXiv:2404.00230, 2024. 4
- [41] midjourney. Midjourney home page. https://www.midjourney.com/home, 2024. Accessed: 2024-10-23.
- [42] Rui Min, Sen Li, Hongyang Chen, and Minhao Cheng. A watermark-conditioned diffusion model for ip protection. *arXiv preprint arXiv:2403.10893*, 2024. 1, 3, 5
- [43] Chong Mou, Xintao Wang, Liangbin Xie, Yanze Wu, Jian Zhang, Zhongang Qi, and Ying Shan. T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 4296–4304, 2024. 1, 2
- [44] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*, 2021. 1, 2
- [45] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International* conference on machine learning, pages 8162–8171. PMLR, 2021. 2, 3
- [46] Ding Sheng Ong, Chee Seng Chan, Kam Woh Ng, Lixin Fan, and Qiang Yang. Protecting intellectual property of generative adversarial networks from ambiguity attacks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3630–3639, 2021. 3
- [47] Gaurav Parmar, Richard Zhang, and Jun-Yan Zhu. On aliased resizing and surprising subtleties in gan evaluation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 11410–11420, 2022.
- [48] Sen Peng, Yufei Chen, Cong Wang, and Xiaohua Jia. Intellectual property protection of diffusion models via the watermark diffusion process. *arXiv* preprint *arXiv*:2306.03436, 2023. 1, 3
- [49] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023. 1, 2
- [50] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 5
- [51] Md Maklachur Rahman. A dwt, dct and svd based watermarking technique to protect the image piracy. arXiv preprint arXiv:1307.3294, 2013. 3, 5
- [52] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. arXiv preprint arXiv:2204.06125, 1 (2):3, 2022. 1, 2

- [53] Jie Ren, Han Xu, Pengfei He, Yingqian Cui, Shenglai Zeng, Jiankun Zhang, Hongzhi Wen, Jiayuan Ding, Pei Huang, Lingjuan Lyu, et al. Copyright protection in generative ai: A technical perspective. arXiv preprint arXiv:2402.02333, 2024. 1
- [54] Ahmad Rezaei, Mohammad Akbari, Saeed Ranjbar Alvar, Arezou Fatemi, and Yong Zhang. Lawa: Using latent space for in-generation image watermarking. arXiv preprint arXiv:2408.05868, 2024.
- [55] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of* the IEEE/CVF conference on computer vision and pattern recognition, pages 10684–10695, 2022. 1, 2
- [56] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the IEEE/CVF conference* on computer vision and pattern recognition, pages 22500– 22510, 2023. 1, 2, 6
- [57] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. Advances in neural information processing systems, 35:36479–36494, 2022. 1, 2
- [58] Chitwan Saharia, Jonathan Ho, William Chan, Tim Salimans, David J Fleet, and Mohammad Norouzi. Image super-resolution via iterative refinement. *IEEE transactions on pattern analysis and machine intelligence*, 45(4):4713–4726, 2022. 2
- [59] sd-vae-ft mse. https://huggingface.co/stabilityai/sd-vae-ft-mse, 2024. 7, 8
- [60] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 5
- [61] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019. 3
- [62] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. arXiv preprint arXiv:2011.13456, 2020. 1
- [63] Matthew Tancik, Ben Mildenhall, and Ren Ng. Stegastamp: Invisible hyperlinks in physical photographs. In *Proceedings* of the IEEE/CVF conference on computer vision and pattern recognition, pages 2117–2126, 2020. 4
- [64] Michael Theis, Randall F Trzeciak, Daniel L Costa, Andrew P Moore, Sarah Miller, Tracy Cassidy, and William R Claycomb. Common sense guide to mitigating insider threats. 2019. 3
- [65] Zhou Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4): 600–612, 2004. 6
- [66] Michael R Watson, Angelos K Marnerides, Andreas Mauthe, David Hutchison, et al. Malware detection in cloud comput-

- ing infrastructures. *IEEE Transactions on Dependable and Secure Computing*, 13(2):192–205, 2015. 3
- [67] Yuxin Wen, John Kirchenbauer, Jonas Geiping, and Tom Goldstein. Tree-rings watermarks: Invisible fingerprints for diffusion images. Advances in Neural Information Processing Systems, 36, 2024. 3
- [68] Cheng Xiong, Chuan Qin, Guorui Feng, and Xinpeng Zhang. Flexible and secure watermarking for latent diffusion model. In Proceedings of the 31st ACM International Conference on Multimedia, pages 1668–1676, 2023. 1, 3
- [69] Ling Yang, Zhilong Zhang, Yang Song, Shenda Hong, Runsheng Xu, Yue Zhao, Wentao Zhang, Bin Cui, and Ming-Hsuan Yang. Diffusion models: A comprehensive survey of methods and applications, 2024. 3
- [70] Zhengyuan Yang, Jianfeng Wang, Zhe Gan, Linjie Li, Kevin Lin, Chenfei Wu, Nan Duan, Zicheng Liu, Ce Liu, Michael Zeng, et al. Reco: Region-controlled text-to-image generation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 14246–14255, 2023. 1, 2
- [71] Shengfang Zhai, Yinpeng Dong, Qingni Shen, Shi Pu, Yuejian Fang, and Hang Su. Text-to-image diffusion models can be easily backdoored through multimodal data poisoning. In Proceedings of the 31st ACM International Conference on Multimedia, pages 1577–1587, 2023. 1, 3
- [72] Kevin Alex Zhang, Lei Xu, Alfredo Cuesta-Infante, and Kalyan Veeramachaneni. Robust invisible video watermarking with attention. arXiv preprint arXiv:1909.01285, 2019.
- [73] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 3836–3847, 2023. 1, 2, 6
- [74] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018. 4
- [75] Shihao Zhao, Dongdong Chen, Yen-Chun Chen, Jianmin Bao, Shaozhe Hao, Lu Yuan, and Kwan-Yee K Wong. Uni-controlnet: All-in-one control to text-to-image diffusion models. Advances in Neural Information Processing Systems, 36, 2024. 1, 2
- [76] Yunqing Zhao, Tianyu Pang, Chao Du, Xiao Yang, Ngai-Man Cheung, and Min Lin. A recipe for watermarking diffusion models. arXiv preprint arXiv:2303.10137, 2023. 1, 2, 3, 5