A Watermark-Conditioned Diffusion Model for IP Protection

Rui Min, Sen Li, Hongyang Chen, Minhao Cheng

Outline

Introduction: Protect Against IP Infringement in the Era of Generative Models

Background: Fingerprinting in Diffusion Models

Problem Setting: Detection and Identification

Methodology: **Wa**termark-conditioned **Diff**usion Model

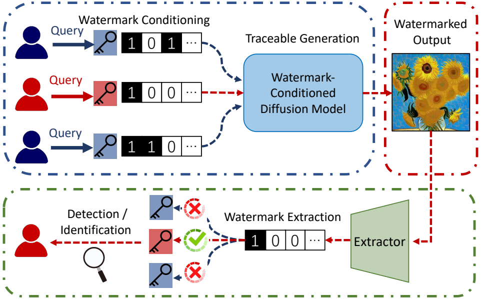
Experimental Results

Ensuring Safe Usage: Generative Models Needs Careful Auditing

- Generative models are powerful tools that affect a wide range of fields, as they create realistic content and drive innovation in various industries [1, 2].
- Although displaying these emergent capabilities, the misuse of the generative model can be harmful coupled with significant ethical and social impacts [3].
- Thus it is urgent to regulate and audit the usage of generative models to make them more responsible and transparent for society.

[1]. Scaling Rectified Flow Transformers for High-Resolution Image Synthesis, ICML 2024
[2]. Improving Image Generation with Better Captions
[3]. A Blueprint for Auditing Generative AI

An Example of How We Protect the IP of A Diffusion Model



- Different users are assigned with unique watermarks, which help regulate the usage of generative models and generative content.

Outline

Introduction: Protect Against IP Infringement in the Era of Generative Models

Background: Fingerprinting in Diffusion Models

Problem Setting: Detection and Identification

Methodology: **Wa**termark-conditioned **Diff**usion Model

Experimental Results

Two Types of Watermarking Schemes

- Post-hoc watermarking after generation.
  - Post-hoc watermarks have been researched for decades and are widely used to protect IP [1]. It is usually model-agnostic and fingerprints the generative content after the generation.
  - While demonstrating both efficacy and robustness, it decouples with the generation process which is more likely to be evaded in practice.
- Implanting watermarks during generation.
  - Recent studies [2, 3] demonstrate the feasibility of fingerprinting during the generation process. This mechanism improves efficiency by eliminating the need to process after the generation process.
  - More importantly, this strategy is hard to bypass due to the integrity of fingerprinting and generation.

[1]. HiDDeN: Hiding Data With Deep Networks, ECCV 2018
[2]. The Stable Signature: Rooting Watermarks in Latent Diffusion Models, ICCV 2023
[3]. Tree-Ring Watermarks: Fingerprints for Diffusion Images that are Invisible and Robust, NeurIPS 2023

Popular Strategies to Watermark Diffusion Models

- DWT-DCT [1]
  - DWT-DCT is adopted in the official implementation of Stable Diffusion.
  - While easy to implement, this watermarking strategy could be easily bypassed by simply commenting a line of code.
- Stable Signature [2]
  - Stable Signature fingerprints the latent decoder of latent diffusion models. Each decoder is assigned a unique watermark and distributed to downstream users.
  - This method is inefficient for a number of users since it needs customized fine-tuning which undermines its efficiency and flexibility.
- Tree-Ring [3]
  - Tree-Ring embeds watermarks into the initial noise by adding a predefined watermarking pattern in the noise's frequency space.
  - This method suffers distinguishing users with different watermarks.

[1]. Digital Watermarking and Steganography
[2]. The Stable Signature: Rooting Watermarks in Latent Diffusion Models, ICCV 2023
[3]. Tree-Ring Watermarks: Fingerprints for Diffusion Images that are Invisible and Robust, NeurIPS 2023

Outline

Introduction: Protect Against IP Infringement in the Era of Generative Models

Background: Fingerprinting in Diffusion Models

Problem Setting: Detection and Identification

Methodology: **Wa**termark-conditioned **Diff**usion Model

Experimental Results

Two Critical Tasks to Evaluate Watermarks

- Detection: Does the image comes from our model?
  - Suppose a user generate an image $\mathbf{p}$ conditioned by watermark $\mathbf{w}_i$ and our task is to determine whether $\mathbf{p}$ is generated from our model.
  - We first extract the source watermark denoted as $\mathbf{w}_s$ and compare it with $\mathbf{w}_i$. We then calculate the number of matched bits as $M_i = \mathbf{w}_s \odot \mathbf{w}_i$, if M_i is larger than a predefined threshold, we can conclude that this image is generated from our model.
- Identification: Who generate this image?
  - Given n users, we have n individual watermarks denoted as $\{\mathbf{w}_1 \dots \mathbf{w}_n\}$, we perform our identification tasks by finding the user with the *closest* watermark with the $\mathbf{w}_s$. Formally, we have:

$$\arg\max_i M_i, \quad i \in \{1 \dots m\}. \tag{1}$$

Outline

Introduction: Protect Against IP Infringement in the Era of Generative Models

Background: Fingerprinting in Diffusion Models

Problem Setting: Detection and Identification

Methodology: **Wa**termark-conditioned **Diff**usion Model

Experimental Results

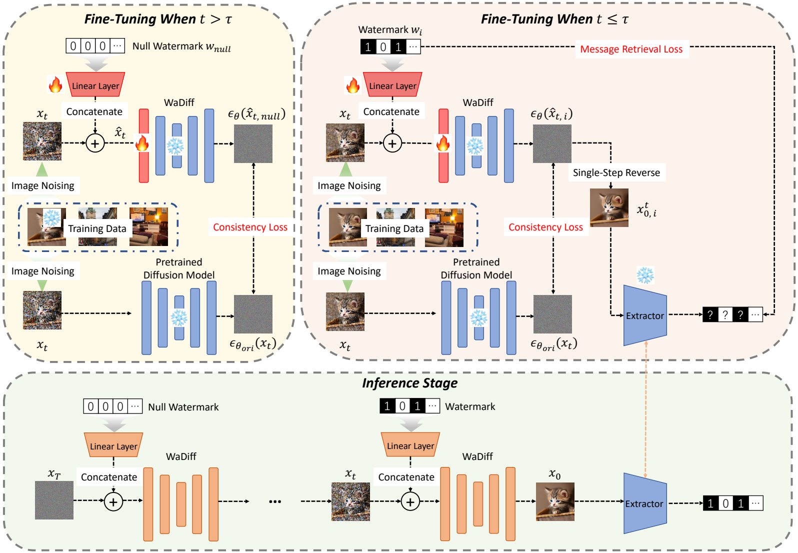
A High-Level Perspective on WaDiff

- Pre-train watermark decoder.
  - We pre-train a StegaStamp decoder in this process and further *freeze* the watermark decoder to guide the fine-tuning of the diffusion model.
  - This process is inspired by the Stable Signature. When designing the WaDiff, we also observe that fine-tuning the diffusion model with a pre-trained watermark decoder is more effective than jointly updating the decoder and the diffusion model.
- Embed watermarks.
  - To fingerprint the generative content, we reverse the noisy latent vector to the initial latent vector at each step and add watermarks to it.
- Preserve image consistency.
  - To further enhance the stealthiness of WaDiff across different users, we regularize the visual appearance of distinct watermarked images to be similar.

How to Fine-tune the Diffusion Model

- Fine-tuning the whole model may significantly affect the generative quality.
  - During the fine-tuning process, instead of updating the whole model, we observe that fine-tuning the first layer is sufficient to embed watermarks.
  - On the contrary, when fine-tuning the whole architecture, we observe an undermined generative performance after a few tuning epochs.
- Add a null watermark when time steps are large.
  - Since the quality of reversed images is low when the time steps are large, we add a null watermark to these stages. The null watermark will never be used when generating images during practical usage.

Overview of WaDiff



Outline

Introduction: Protect Against IP Infringement in the Era of Generative Models

Background: Fingerprinting in Diffusion Models

Problem Setting: Detection and Identification

Methodology: **Wa**termark-conditioned **Diff**usion Model

Experimental Results

Identification Performance against up to One Million Users

Table 1: This table includes our main results. Trace m indicates the tracing accuracy (%) of our identification among m users in total.

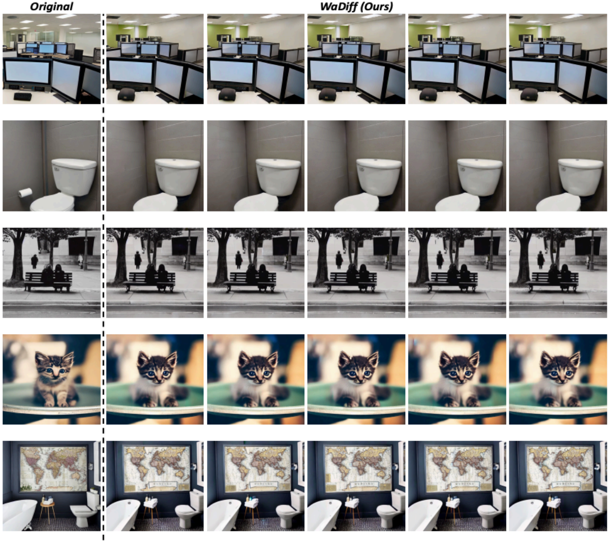
Model	Type	Method	AUC	Trace 10^4	Trace 10^5	Trace 10^6	Trace Avg	SSIM(↑)	FID Diff(↓)
Stable Diffusion	Post Generation	DwtDct	0.917	76.30	74.70	72.90	74.63	0.999	-0.36
		StegaStamp	1.000	99.98	99.98	99.96	99.97	0.999	+0.27
	Merged Generation	Tree-Ring$_{Rand}$	0.999	0.04	0.00	0.00	0.01	0.457	+0.14
		Tree-Ring$_{Rings}$	0.999	0.00	0.00	0.00	0.00	0.575	+0.77
		WaDiff (Ours)	0.999	98.20	96.76	93.44	96.13	0.999	+0.41
256×256 ImageNet	Post Generation	DwtDct	0.936	71.30	68.10	65.20	68.20	0.997	-0.05
		StegaStamp	1.000	99.98	99.98	99.98	99.98	0.998	+0.11
	Merged Generation	Tree-Ring$_{Rand}$	0.999	0.00	0.00	0.00	0.00	0.584	+0.17
		Tree-Ring$_{Rings}$	0.999	0.00	0.00	0.00	0.00	0.652	+0.23
		WaDiff (Ours)	1.000	99.68	99.38	98.78	99.28	0.997	+0.08

Robustness Analysis

Table 2: This table reports WaDiff tracing accuracy (%) and AUC under diverse data augmentations.

Model	Case	Resize	Blurring	Color Jitter	Noising	JPEG	Combine	Avg
Stable Diffusion	AUC	0.999	0.999	0.999	0.997	0.999	0.999	0.999
	Trace 10^4	97.02	97.14	96.00	88.52	93.48	93.02	94.19
	Trace 10^5	94.34	94.12	88.56	81.14	87.66	84.26	88.34
	Trace 10^6	89.46	87.40	82.14	72.50	80.30	78.04	81.64
256×256 ImageNet	AUC	0.999	0.999	0.999	0.999	0.999	0.999	0.999
	Trace 10^4	98.90	94.48	98.56	91.80	92.06	91.88	94.61
	Trace 10^5	97.78	89.90	96.48	84.46	88.70	85.74	90.51
	Trace 10^6	96.02	82.42	94.50	76.26	77.88	76.88	83.99

Examples of Watermarked Images (COCO)



Examples of Watermarked Images (ImageNet)