Towards Robust Content Watermarking Against Removal and Forgery Attacks

Yifan Zhu *
AMSS and UCAS, CAS †
zhuyifan@amss.ac.cn

Yihan Wang * ‡
University of Waterloo
yihan.wang@uwaterloo.ca

Xiao-Shan Gao ‡
AMSS and UCAS, CAS †
xgao@mmrc.iss.ac.cn

Abstract

*Generated contents have raised serious concerns about copyright protection, image provenance, and credit attribution. A potential solution for these problems is watermarking. Recently, content watermarking for text-to-image diffusion models has been studied extensively for its effective detection utility and robustness. However, these watermarking techniques are vulnerable to potential adversarial attacks, such as removal attacks and forgery attacks. In this paper, we build a novel watermarking paradigm called Instance-Specific watermarking with Two-Sided detection (**ISTS**) to resist removal and forgery attacks. Specifically, we introduce a strategy that dynamically controls the injection time and watermarking patterns based on the semantics of users' prompts. Furthermore, we propose a new two-sided detection approach to enhance robustness in watermark detection. Experiments have demonstrated the superiority of our watermarking against removal and forgery attacks.*

1. Introduction

Diffusion models have become the dominant class of generative models for visual tasks, powering many popular text-to-image and text-to-video systems such as Stable Diffusion [47], DALL-E [46], Midjourney, and Sora [39]. Content produced by these models has become widespread across social media and even traditional publications [36]. Notably, a synthetically generated image has even won a world photography award [16].

The photorealistic quality of such generated media has raised serious concerns about intellectual property protection, including issues of image provenance, authorship, and credit attribution. Watermarking has emerged as a promising solution for verifying whether an image originates from a specific diffusion model [17, 58, 61]. In this approach, identity markers are embedded into the generated outputs during the synthesis process without degrading their semantics or perceptual quality. During detection, model providers can confirm the provenance of an image by detecting or recovering these markers. Researches on watermarking for diffusion models have grown rapidly in recent years, with many works focusing on improving detection accuracy, image fidelity, and robustness against natural variations such as transformations, noise, or compression [9, 56, 66].

However, since ownership and credit for generated content can carry reputational and even financial implications, malicious adversaries have strong incentives to circumvent watermarking mechanisms. Such attackers may attempt to remove identity markers from watermarked images or to forge watermarked images from unmarked ones. These attacks often exploit gradients propagated through the diffusion process and the statistical properties of large sets of watermarked samples, achieving high success rates in watermark removal and forgery [26, 29, 41, 60]. Such vulnerability presents severe challenges to the reliability of current watermarking methods. To address these challenges, we propose *Instance-Specific watermarking with Two-Sided detection* (**ISTS**), which achieves enhanced resilience against forgery and removal attacks while maintaining comparable utility in non-adversarial settings.

Our key insight is that, despite existing attacks being frequently characterized by black-box access, in which the adversary does not know the underlying generative model or watermarking approach, the prevailing use of static, single-type watermarking methods effectively grants the attacker additional prior knowledge. For instance, the well-known Tree-Ring watermarking [58] injects a fixed ring pattern into the Fourier space of latent noises for all prompts. This unintended leakage allows adversaries to more easily infer the watermark's structural characteristics, facilitating its removal or the creation of forged watermarks. To mitigate this vulnerability, we propose a strategy that customizes both the watermark pattern and the injection time for each generation. Concretely, given a prompt, we first generate a non-watermarked image and encode it into a semantic feature vector using the CLIP encoder. We then determine the wa-

*These authors contributed equally to this work.
†Academy of Mathematics and Systems Science, Chinese Academy of Sciences and University of Chinese Academy of Sciences.
‡Corresponding authors.

termarking parameters, such as the pattern position and the injection time step, based on this semantic vector through a pretrained semantic-based selector. Finally, using the selected parameters, we generate the corresponding watermarked image from the same prompt. For detection, we first encode the suspicious image into its semantic feature vector, retrieve the corresponding watermarking parameters, and then compare the target region with the predefined watermark pattern. Since watermark injection minimally affects image semantics, the semantic features of watermarked and non-watermarked images remain closely aligned, ensuring that the recovered parameters are consistent with those used during the watermark injection process.

In addition, we observe that the commonly adopted one-sided detection introduces a critical vulnerability to removal attacks. To address this issue, we propose a two-sided detection scheme that captures opposite latent representations, effectively strengthening robustness against such attacks while preserving the watermark's utility.

We conduct experiments on various content watermarking with our ISTS against three representative removal attacks and three forgery attacks. Evaluation results demonstrate the superior performance of our proposed ISTS, achieving state-of-the-art detection AUC and TPR@1%FPR across removal and forgery attacks in both average and worst-case scenarios.

2. Related Works

Watermarking. Watermarking on generative models has been widely studied in recent years. In text-to-image generative models, a representative watermarking method called Tree-Ring [58] injects a specific ring pattern into the Fourier space of the latent representation, achieving higher robustness compared with traditional image watermarking methods [10, 54, 64]. Followed up with many content watermarking methods, such as ZoDiac [66], RingID [9], ROBIN [25], Gaussian-Shading [61], and Shallow Diffuser [34]. More works [2, 18, 43, 56, 62] have been proposed to further enhance content watermarking in diffusion models. Other works try to inject watermarking into text-to-image generative models by finetuning the model [53, 65, 71] or training a watermarking extractor [14]. Gunn et al. [17] tries to use pseudorandom error-correcting codes to inject an undetectable watermark into generative image models. SEAL [3] Using SimHash to control the randomness of the denoising process. In large language models, Kirchenbauer et al. [30] first exploits the Red-Green list to imprint watermarks into the generated text. Further research on watermarking for language models has also been studied in recent years [8, 31, 32, 37, 59, 69]. More broadly, watermarking in text-to-video generative models has been investigated recently [15, 22, 23]; watermarking in generative tabular data and data poisoning attacks has also been considered [19, 72, 73].

Removal and forgery attacks on watermarking. Removal attacks have been widely studied in the area of post-hoc image watermarking, including distortion methods [1, 55], reconstruction and generation processes [6, 35, 38, 69], and learning-based approaches [24, 27, 40]. Forgery attacks against traditional image watermarking have also been investigated [12, 33, 49, 55]. Ba et al. [4] utilizes channel-aware feature extraction to remove and forge image watermarks. Content watermarking shows strong robustness against image distortion and diffusion purification [42, 63], however, a recent study has found that content watermarking is vulnerable to removal and forgery attacks, even under simple averaging [60]. Müller et al. [41] utilizes a surrogate diffusion model to remove content watermarking by optimizing the watermarked latent in the opposite direction and to forge content watermarking by approximating the watermark pattern in the latent space when a single watermarked image is leaked. Jain et al. [26] achieves removal and forgery attacks via the VAE encoder with a single watermarked image. Recent works [5, 28, 68] have demonstrated that watermarking in large language models is also fragile to removal (scrubbing) and forgery (spoofing) attacks.

In this paper, we mainly focus on content watermarking in text-to-image generative models, which have been well studied and are vulnerable to removal and forgery attacks [26, 41, 60]. We introduce a novel watermarking method, ISTS, to improve robustness against these attacks.

3. Background

3.1. Preliminaries

Diffusion Models. Diffusion models [21, 52] have achieved remarkable success in the area of image generation. A representative framework called the Denoising Diffusion Probabilistic Model (DDPM) [21] progressively adds Gaussian noise $\epsilon \sim \mathcal{N}(0, I)$ to the original image x_0 through the following forward process:

$$x_t = \sqrt{\alpha_t} x_0 + \sqrt{1-\alpha_t}\epsilon \tag{1}$$

with a certain noise schedule $\{\alpha_t\}$. We approximate the reverse process by estimating the noise ϵ_t using the diffusion model. To enhance sample efficacy, Song et al. [51] proposed Denoising Diffusion Implicit Models (DDIM) to predict the previous state x_{t-1} by:

$$x_{t-1} = \sqrt{\alpha_{t-1}}\left(\frac{x_t - \sqrt{1-\alpha_t}\epsilon_\theta(x_t, t)}{\sqrt{\alpha_t}}\right) + \sqrt{1-\alpha_{t-1}}\epsilon_\theta(x_t, t), \tag{2}$$

where ϵ_θ is parameterized and trained to predict ϵ_t at time t. A merit of DDIM is that one can sample x_t from x_{t-1} through the DDIM inversion process, similar to Eq. (2).

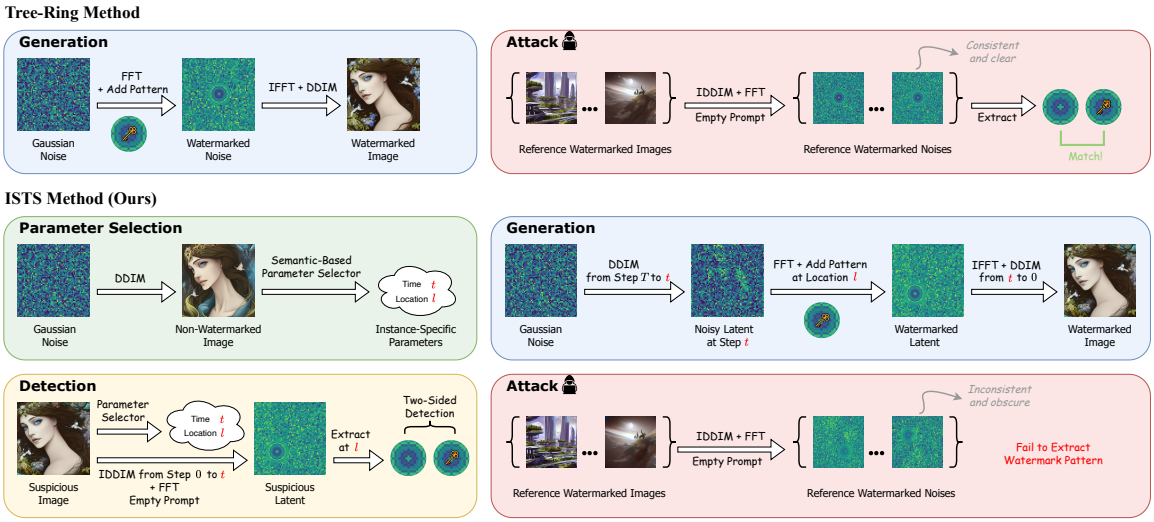

Figure 1. Overview of our ISTS method. The **top** row shows the original Tree-Ring watermarking. To generate watermarked images, it injects tree-ring patterns in the center of the frequency domain at the initial noisy space. Such a static scheme leaks information about the watermark pattern and thus exposes high vulnerability to removal and forgery attacks. The **bottom** row shows our dynamic approach. For each generation, we first generate a non-watermarked image and decide time and location parameters (t, l) using *a semantic-based selector*. Then, we execute the first $T - t$ steps of DDIM, inject watermarks at coordinate l in the frequency domain, and continue the rest of DDIM generation. During detection, we retrieve the parameters from a suspicious image and extract the watermarking area, followed by *two-sided detection*. In adversarial settings, an attacker unaware of time and location parameters fails to extract the ground-truth watermark patterns from reference images. Thus, the proposed approach achieves enhanced robustness against removal and forgery attacks.

Text-to-Image (T2I) Diffusion Models. Diffusion models can be generalized to text-to-image [13, 48] by embedding image x into the latent space with the encoder $\mathcal{E}$ to obtain latent $z_0 = \mathcal{E}(x)$, then guiding the image generation process with a text prompt p. We define the denoising process to be

$$z_{t_1} = \mathcal{M}_{t_2 \to t_1}(p, z_{t_2}),$$

where $\mathcal{M}_{t_2 \to t_1}$ is to denoise from step t_2 to t_1 with the latent diffusion model $\mathcal{M}$ when $t_1 < t_2$. After obtaining the denoised latent $\tilde{z}_0$, decode the latent using the decoder $\mathcal{D}$ to obtain the generated image $\tilde{x}_0 = \mathcal{D}(\tilde{z}_0)$. To simplify the notation, we omit the decoder $\mathcal{D}$ in this paper when no ambiguity exists. For the inversion process, similarly, we denote it as

$$z_{t_2} = \mathcal{M}^{\text{Inv}}_{t_1 \to t_2}(p, z_{t_1}), t_1 < t_2. \tag{3}$$

The classifier-free guidance (CFG) [20] has also been employed for the sampling process.

Content watermarking. A well-known content watermarking technique in text-to-image diffusion models, called Tree-Ring [58], injects a watermark pattern into the initial latent noise vector z_T. Following Tree-Ring, many content watermarking methods have been developed, including Gaussian-Shading [61], ROBIN [25], RingID [9], ZoDiac [66], Shallow Diffuse [34], etc. These methods modify the generation process of text-to-image diffusion models without finetuning, achieving high robustness compared to existing image watermarking methods and providing a plug-and-play approach to inject effective watermarking.

Watermarking detector. The watermark detector D is a binary classifier, where $D(x) = 1$ if the image x is recognized as a watermarked instance and $D(x) = 0$ if x is recognized as a non-watermarked instance.

3.2. Threat Model of Removal and Forgery Attacks

Adversary's objectives. In our scenario, the adversary has two objectives, removal attacks and forgery attacks.

Removal Attacks: In this setting, the adversary starts with a watermarked image generated by a T2I model. The objective is to slightly modify the image so that the resulting output bypasses watermark detection while preserving perceptual similarity. Formally, for a watermarked image I^w, removal attacks try to find the perturbation δ, such that

$$D(I^w + \delta) = 0, \;\; d(I^w + \delta, I^w) \leq \epsilon,$$

where d is a metric measuring the difference of two images.

Forgery Attacks: The adversary begins with a benign (non-watermarked) image and seeks to alter it such that it is falsely recognized as watermarked during detection while preserving perceptual similarity. Formally, given a benign image I^b, the forgery attacker aims to search for a perturbation δ, such that

$$D(I^b + \delta) = 1, \;\; d(I^b + \delta, I^b) \leq \epsilon.$$

Adversary's knowledge and capability. The adversary is assumed to have no access to the weights of the target diffusion model, and is unaware of the watermarking algorithm or its implementation details. However, they can leverage surrogate models to compute gradients, granting a strong capability that enables a range of gradient-based attacks [41], and can possess a reference image containing a watermark, which could be used in attempts to forge or replicate the watermark. Furthermore, they may obtain some watermarked images to extract population characteristics [60].

3.3. Failure of Existing Content Watermarking Against Removal and Forgery Attacks

Recent works [26, 41, 60] have shown that content watermarking is susceptible to removal and forgery attacks, even if only one watermarked image is leaked, pointing to significant risks associated with the real-world deployment of content watermarking. For example, as we showed in Figure 2, existing content watermarking can be almost completely destroyed by the removal and forgery attacks proposed by Müller et al. [41]. After the removal attack, the detector can achieve only less than 0.1 AUC between benign generated images and watermarked images under removal for Gaussian-Shading, ROBIN, RingID, and Zodiac watermarking, rendering them ineffective. After the forgery attack involving only one leaked watermarked image, benignly generated images can be modified into watermarked images, and the watermarking detector will confidentially view these forged images as watermarked images, with an AUC of nearly 1.0. The failure of existing content watermarking facilitates our design of a new watermarking paradigm that is robust against such forgery and removal attacks.

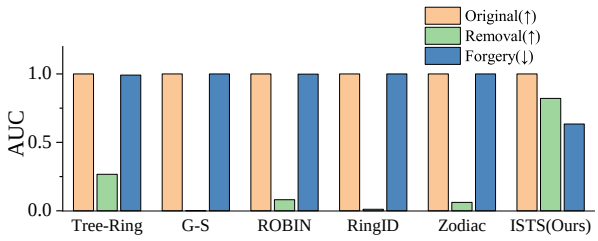

Figure 2. The detection AUC of benign watermarking, and watermarking after removal and forgery attacks [41]. Results demonstrate that existing methods are vulnerable to removal attacks (lower AUC) and forgery attacks (higher AUC), while our ISTS is robust against them.

4. Methodology

To overcome the vulnerability of watermarking against removal and forgery attacks, we propose a new watermarking method called *Instant-Specific watermarking with Two-Sided detection* (**ISTS**), which customizes both the watermark location and the injection time for each instance during generation, and conducts a two-sided test for detection.

4.1. Instance-Specific Watermarking

Existing content watermarking methods, such as Tree-Ring, RingID, and Zodiac, rely on a static watermarking pattern to assist in detection. The pattern is injected into the initial noisy latent space through a Fourier transformation, making it invisible to humans and gaining decent robustness under image distortions. However, such static, single-type watermarking grants potential attackers additional prior knowledge about the scheme. This unintended leakage enables them to infer the watermark's structural characteristics, facilitating removal and forgery attacks. For instance, an adversary can reverse images to the latent space using existing methods, such as DDIM-Inversion [11]. Therefore, with a watermarked image for reference, the adversary can forge more watermarked images from benign ones by minimizing the distance between the reference and benign latent vectors with gradient descent [41].

Dynamic patterns and injections. To enhance resilience to removal and forgery attacks, we introduce instance-specific parameters to control the watermark injected for each image generation, thereby increasing the variety of watermarks and limiting the efficacy of the adversary's operation. For each image generation, we dynamically select parameters in both time and location dimensions, i.e., the injection step t and the pattern coordinate l, making it substantially harder for an adversary to predict or exploit them.

Specifically, as shown in Figure 1, we first generate a non-watermarked image, extract its features using a CLIP encoder, and map those features to time and location parameters (t, l) through a semantic-based selector. Then, the watermark pattern is injected at coordinate l in step t, as shown in Algorithm 2 and Appendix A.1. The parameter selector is obtained through K-Means clustering applied to non-watermarked images, as demonstrated in Algorithm 1, in which the parameter mapping ϕ is predefined based on the modulo operation, which will be demonstrated in Appendix A.1. Since the semantics of non-watermarked and watermarked images are similar, the paired images tend to be predicted to the same parameters, resulting in consistency for generation and detection, thereby ensuring strong detection performance even under adversarial conditions.

Dynamic patterns and injections significantly enhance the correlation between image concepts and watermarking styles, further degrading the existing removal and forgery attacks that use a single watermarked image [26, 41] or the averaging of watermarked images [60] as features to extract the watermarking pattern for each generated image.

Resilience improvement. Recall that the adversary lacks knowledge of the watermarking algorithm's internal design

and must perform a single-shot attack under an assumed standard setting. Thus, the greater the discrepancies between the assumed and the actual configurations, the lower the attack's success rate. To illustrate how our proposed scheme benefits the resilience to removal and forgery attacks, we will analyze six concrete attacks.

Regarding forgery attacks, Müller et al. [41] and Jain et al. [26] both consider the scenario in which an attacker has access to a single watermarked image for reference. Our approach requires the watermark to be placed according to the outputs of a semantic-based selector. However, in these two attacks, the watermark pattern forged from the reference image does not match the semantics of the benign image, thereby resulting in a detection failure. Moreover, they attempt to align the reference feature and the forged feature in the latent space. However, the dynamic time parameter hinders the attack from tracing back to the exact injection step at which gradient-based optimization is conducted. Yang et al. [60] considers a forgery attacker who obtains a multitude of non-watermarked and watermarked images. The attacker extracts watermarking features by calculating the residuals of the averaged watermarked images and the averaged non-watermarked images. Our proposed scheme diversifies the watermarking features residing in the pixel space. Thus, the features in one watermarked image might offset those in other watermarked images, and their average fails to present meaningful information about any watermarking pattern.

Regarding removal attacks, Jain et al. [26] uses a plain image with all values equal to the mean of the targeted watermarked image as the object of optimization. However, this results in biased removal features, since our dynamic patterns generate distinct watermark features across different semantics, thereby substantially degrading their attack performance. In addition, the gradient-based removal attack by Müller et al. [41] suffers from an injection-step mismatch issue, similar to its forgery counterpart. Furthermore, in Yang et al. [60], the residuals between averaged non-watermarked and watermarked images no longer capture watermark features under our dynamic pattern. Consequently, their strategy of removing watermarks by subtracting the average fails to remain effective.

4.2. Two-Sided Detection

Beyond the generation process, we find that the detection process could also expose the vulnerabilities of watermarking. Existing methods like Tree-Ring variants, typically rely on one-sided detection with the detection metric

$$d = \frac{1}{|M|} \sum_{i \in M} |W_i - \mathcal{F}(z_T)_i|,$$

in that W_i is the watermarking pattern under dimension index i and $z_T = \mathcal{M}^{\text{Inv}}_{0 \to T}(z_0)$ for the suspect image latent

Algorithm 1 Parameter Selector Training

Input: Model $\mathcal{M}$, prompt set $\mathcal{P}$, CLIP extractor g, clustering number N, a parameter mapping ϕ.
Output: Parameter selector f.
for p in $\mathcal{P}$ **do**
  Sample noise z_T from $\mathcal{N}(0, I)$.
  $I_p^c \leftarrow \mathcal{M}(p, z_T)$. ▷ Obtain non-watermarked images
  $z_p^c \leftarrow g(I_p^c)$. ▷ Extract features
$y_p^c \leftarrow \text{K-Means}(z_p^c, \{z_p^c\}_{p \in \mathcal{P}}, N)$. ▷ Assign labels
Train a classifier h using labeled dataset $\{(z_p^c, y_p^c)\}_{p \in \mathcal{P}}$.
return $f \leftarrow \phi \circ h \circ g$.

$z_0 = \mathcal{E}(x_0)$ under DDIM Inversion process [11] as proposed in Tree-Ring [58], $\mathcal{F}$ is the Fourier transformation. This detection approach introduces a critical vulnerability to removal attacks. Specifically, an adversary can eliminate the watermark by optimizing the generated images to induce opposite latent representations (Details in Appendix A.2). A simple yet effective removal attack proposed by Müller et al. [41] has shown significant degradation of watermarking, as outlined in Table 1. It induces the watermarking latents into the opposite to generate the removed images. To overcome this issue, we modify our detection approach from one-sided to two-sided. With this simple yet effective change, our watermarking paradigm demonstrates stronger robustness against removal attacks while preserving detection efficacy in non-adversarial conditions. Formally, the detection metric works as follows:

$$d = \min\left\{ \frac{1}{|M|} \sum_{i \in M} |W_i - \mathcal{F}(z_T)_i|, \frac{1}{|M|} \sum_{i \in M} |W_i + \mathcal{F}(z_T)_i| \right\}.$$

It is noteworthy that two-sided detection will not affect the evaluation metric of the original non-watermarked latents z_T, as $\mathcal{F}(z_T)_i$ is a standard Gaussian distribution, which is symmetric with respect to sign changes. For watermarked latents, the matched pattern can still be extracted as we introduce the minimum operation. Therefore, two-sided detection is able to maintain detection accuracy when removal attacks are not engaged.

5. Experiments

5.1. Experimental Setup

Datasets and models. In this work, we deploy Stable-Diffusion-2-1-base [48] as the text-to-image diffusion model and use [50] as our text prompts to ensure a fair comparison with prior work [3, 58]. Following the setting from Müller et al. [41], we select 100 pairs of watermarked and non-watermarked generated images for the evaluation of removal and forgery attacks. In non-adversarial scenarios, we measure on 1,000 pairs of images.

Algorithm 2 Watermark Injection

Input: Model $\mathcal{M}$, prompt p, pattern W, parameter selector f, total generation step T.
Output: Generated watermarked image I^w.
Sample noise z_T from $\mathcal{N}(0, I)$.
$I^c \leftarrow \mathcal{M}(p, z_T)$. ▷ Generate a non-watermarked image
$(t, l) \leftarrow f(I^c)$ ▷ Select time and location
$z_t \leftarrow \mathcal{M}_{T\to t}(p, z_T)$. ▷ Denoise to step t
$z_t^w \leftarrow z_t \oplus \text{Offset}(W, l)$ ▷ Add at coordinate l
$I^w \leftarrow \mathcal{M}_{t\to 0}(p, z_t^w)$. ▷ Denoise to image

Algorithm 3 Watermark Detection

Input: Suspicious image I, model $\mathcal{M}$, encoder $\mathcal{E}$, parameter selector f, watermarking pattern W, threshold τ, watermarking mask M.
$z_0 \leftarrow \mathcal{E}(I)$, $(t, l) \leftarrow f(I)$, $z_t \leftarrow \mathcal{M}^{\text{Inv}}_{0\to t}(z_0)$.
$W \leftarrow \text{Offset}(W, l)$
$d \leftarrow \min \frac{1}{|M|} \left\{ \sum_{i\in M} |W_i - \mathcal{F}(z_{t,i})|, \sum_{i\in M} |W_i + \mathcal{F}(z_{t,i})| \right\}$
▷ Two-Sided Detection
if $d < \tau$ **then**
  return Watermarked
else
  return Non-Watermarked

Baselines. We compare ISTS with recent content watermarking methods, including Tree-Ring [58], Shallow Diffuse [34], Gaussian-Shading [61], ROBIN [25], RingID [9], ZoDiac [66], and SEAL [3]. For removal and forgery attacks, we use imprinting attacks with gradient descent [41] (Imp-Removal/Forgery), simple averaging attacks [60] (Avg-Removal/Forgery), and VAE attacks with gradient descent [26] (VAE-Removal/Forgery). Details of these attacks are provided in Appendix A.1.

Metrics. We deploy the area under the curve (AUC) of the receiver operating characteristic (ROC) curve, and the True Positive Rate (TPR) when the False Positive Rate (FPR) is 1% (TPR@1%FPR) as our detection metrics. For removal attacks, the detection metrics are evaluated between benign non-watermarked images and watermarked images after removal attacks. For forgery attacks, the detection metrics are evaluated between benign non-watermarked images and non-watermarked images after forgery attacks. Therefore, watermarking is more robust against removal attacks if the AUC and TPR@1%FPR are higher, and is more robust against forgery attacks if the AUC and TPR@1%FPR are lower. In practice, the detection threshold is fixed after the watermarking paradigm is proposed; thus, we utilize benign images as the non-watermarked criterion to maintain a consistent detection threshold for the TPR@1%FPR metric.

5.2. Main Results

Results against removal attacks. We evaluate the detection AUC and TPR@1%FPR with existing baseline watermarking, including Tree-Ring, Shallow Diffuse, Gaussian-Shading, ROBIN, RingID, Zodiac, SEAL, and our proposed ISTS, against three existing removal attacks against T2I diffusion watermarking: Imp-Removal, Avg-Removal, and VAE-Removal. The detailed results are provided in Table 1. All these watermarking methods show strong detection performance when no removal attacks are involved (Original). However, after removal attacks, different watermarking methods demonstrate different utilities, where our ISTS achieves superior performance.

In detail, for the Imp-Removal attack, almost all existing watermarking methods are severely degraded; the strongest baseline, Shallow Diffuse, only obtains an AUC of less than 0.7. Our watermarking improves the robustness with a significant gap, achieving over a 20% AUC improvement (from 0.675 to 0.821) and an 18-fold TPR@1%FPR enhancement (from 0.01 to 0.18). For Avg-Removal, Shallow Diffuse, SEAL, and our ISTS show strong robustness, where our method achieves the best AUC and TPR@1%FPR. VAE-Removal is a relatively weak removal attack against which many existing watermarking techniques are robust, including Gaussian Shading and RingID, which are particularly effective. ISTS also demonstrates comparable robustness in the VAE-Removal attack.

We also make a general evaluation of these representative removal attacks with an average scenario and a worst-case scenario. In an average scenario, we assume that attackers do not have prior knowledge of these removal attacks against watermarking and take a random selection of the attack method against a certain watermarking. In this case, we evaluate the detection performance by calculating the average AUC and TPR@1%FPR. Our ISTS achieves the best robustness in this scenario, with over 0.93 AUC and 0.58 TPR@1%FPR. In the worst-case scenario, we assume that attackers can choose the best removal method against certain watermarking, and the detection metrics are evaluated by finding the lowest AUC and TPR@1%FPR across these attacks. In this scenario, we find that Imp-Removal dominates, and the superior performance of ISTS against Imp-Removal induces the best robustness in the worst-case scenario.

Results against forgery attacks. Similar to removal attacks, we also evaluate the detection AUC and TPR@1%FPR for various content watermarking methods against three existing forgery attacks, Imp-Forgery, Avg-Forgery, and VAE-Forgery. The detailed results are provided in Table 2. It is noteworthy that, unlike removal attacks, lower AUC/TPR@1%FPR indicates better watermarking against forgery attacks, as the watermarked images are harder to be forged. For Imp-Forgery, all exist-

Table 1. The detection metric (AUC/TPR@1%FPR) of various watermarking methods against removal attacks. "Average" means the average AUC/TPR@1%FPR across three removal attacks, Imp-Removal, Avg-Removal and VAE-Removal. "Worst-Case" means the worst performance against these removal attacks, i.e., the lowest AUC and TPR@1%FPR under a certain attack. Our ISTS demonstrates the strongest robustness against Imp-Removal and Avg-Removal, and outperforms all existing baselines in both averaged and worst-case scenarios for removal attacks.

Watermarking Method	Original	Imp-Removal	Avg-Removal	VAE-Removal	Average	Worst-Case
Tree-Ring	0.9999/1.00	0.2672/0.00	0.5266/0.08	0.9728/0.34	0.5889/0.14	0.2672/0.00
Shallow Diffuse	1.0000/1.00	0.6752/0.00	0.9730/0.49	0.9987/0.93	0.8823/0.47	0.6752/0.00
Gaussian Shading	1.0000/1.00	0.0000/0.00	0.3707/0.01	**1.0000/1.00**	0.4569/0.34	0.0000/0.00
ROBIN	1.0000/1.00	0.0815/0.00	0.7415/0.00	0.9610/0.58	0.5947/0.19	0.0815/0.00
RingID	1.0000/1.00	0.0121/0.01	0.4035/0.16	**1.0000/1.00**	0.4719/0.39	0.0121/0.01
Zodiac	0.9999/1.00	0.0625/0.00	0.2558/0.00	0.7901/0.04	0.3695/0.01	0.0625/0.00
SEAL	0.9998/1.00	0.5078/0.00	0.9590/0.65	0.7884/0.37	0.7517/0.34	0.5078/0.00
ISTS(Ours)	1.0000/1.00	**0.8210/0.18**	**0.9900/0.70**	0.9979/0.85	**0.9363/0.58**	**0.8210/0.18**

Table 2. The detection metric (AUC/TPR@1%FPR) of various watermarking methods against forgery attacks. "Average" and "Worst-Case" mean the average and the lowest AUC/TPR@1%FPR across three forgery attacks, Imp-Forgery, Avg-Forgery and VAE-Forgery, respectively. Our ISTS demonstrates the strongest robustness against Imp-Forgery, and outperforms all existing baselines in both averaged and worst-case scenarios for forgery attacks.

Watermarking Method	Original	Imp-Forgery	Avg-Forgery	VAE-Forgery	Average	Worst-Case
Tree-Ring	0.9999/1.00	0.9914/0.73	0.7932/0.01	0.9525/0.48	0.9124/0.41	0.9914/0.73
Shallow Diffuse	1.0000/1.00	0.7621/0.11	0.6285/0.02	0.9533/0.55	0.7813/0.23	0.9533/0.55
Gaussian Shading	1.0000/1.00	1.0000/1.00	0.4702/0.00	1.0000/1.00	0.8234/0.67	1.0000/1.00
ROBIN	1.0000/1.00	0.9995/0.95	0.8784/0.03	0.9290/0.29	0.9356/0.42	0.9995/0.95
RingID	1.0000/1.00	1.0000/1.00	1.0000/1.00	1.0000/1.00	1.0000/1.00	1.0000/1.00
Zodiac	0.9999/1.00	1.0000/1.00	**0.2436/0.00**	0.9541/0.42	0.7326/0.47	1.0000/1.00
SEAL	0.9998/1.00	0.9536/0.48	0.4331/0.00	**0.7213/0.04**	0.7027/0.17	0.9536/0.48
ISTS(Ours)	1.0000/1.00	**0.6340/0.00**	0.4737/0.00	0.9491/0.37	**0.6856/0.12**	**0.9491/0.37**

ing methods except Shallow Diffuse are easily forged (over 0.95 AUC), and ISTS obtains superior performance, achieving over 20% improvement on AUC compared with the strongest baseline, Shallow Diffuse (0.76 to 0.63). Avg-Forgery is a relatively weak attack, with almost all existing watermarking (except RingID) having good resistance against it. Zodiac performs best, and ISTS also gets comparable results. VAE-Forgery seems stronger against all existing methods, SEAL dominates this attack, ISTS achieves comparable results with some baselines like Tree-Ring, Shallow Diffuse, ROBIN and Zodiac.

In the average and worst-case scenarios, our ISTS still demonstrate superiority across three forgery attacks, with the lowest AUC (0.685 and 0.949 for average and worst-case scenarios) and TPR@1%FPR (0.12 and 0.37 for average and worst-case scenarios).

It is noteworthy that, although SEAL outperforms against VAE-Forgery and achieves comparable results with our ISTS in the average and worst-case scenarios, their poor image quality and robustness against image distortions and purifications will limit their applicability, as demonstrated in the next section.

5.3. More Experimental Results

Image quality and semantics. We evaluate the generated image quality with metrics including Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index (SSIM) [57], and Learned Perceptual Image Patch Similarity (LPIPS) [67]. Furthermore, we also measure the CLIP-Score [45] between generated images and prompts using OpenCLIP-ViT/G [7]. Results shown in Figure 3 demonstrate that for PSNR, SSIM, and LPIPS, our ISTS outperforms various content watermarking methods like Tree-ring, Gaussian-Shading, RingID, and SEAL, and achieves comparable results with ROBIN. For CLIP-Score, existing methods have similar results, implying good semantic alignment with text prompts. It is worth noting that, although ROBIN has comparable image quality with ISTS, it is very fragile to removal and forgery attacks as displayed in Tables 1 and 2.

Robustness against image distortions and purifications. To further investigate the robustness of content watermarking, followed the setting of Wen et al. [58], we evaluate them on several well-known image distortions, including 75° rotation (Rotation), $\sigma = 0.1$ Gaussian noise (Noise), Gaussian blur with 8×8 filter size (Blurring), 75% ran-

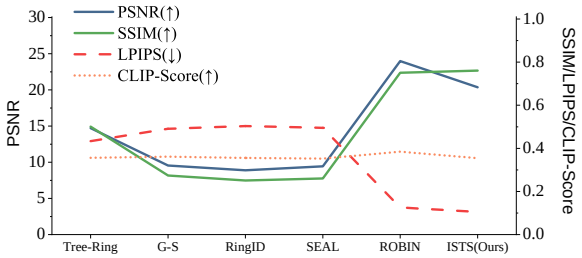

Figure 3. The PSNR, SSIM, LPIPS and CLIP-Score for various content watermarking. Our ISTS achieves comparable results with ROBIN and outperforms other methods. G-S means Guassian-Shading.

dom cropping and scaling (Cropping), and 25% JPEG compression (JPEG). Moreover, we also evaluate the effectiveness of watermarking under reconstruction through diffusion purification [70] (Diffpure). Results in Figure 4 show that Tree-Ring, RingID ,and our ISTS obtain decent robustness against these image distortions. Gaussian-Shading is vulnerable to Rotation, ROBIN is fragile to Diffpure, and SEAL becomes ineffective under Rotation, Blurring, Cropping, and Diffpure.

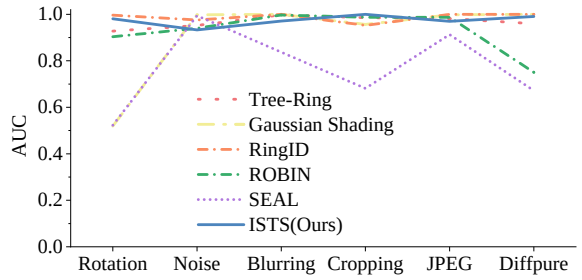

Figure 4. Watermarking detection AUC under various image distortions (Rotation, Noise, Blurring, Cropping, JPEG) and diffusion purification (Diffpure). Results demonstrate that our method achieves high AUC across these data augmentations.

To further quantify the robustness for watermarking, we provide the detection AUC in both average and worst-case scenarios across these distortions on Table 3. ISTS possesses comparable robustness with Tree-Ring and RingID, and outperforms Gaussian Shading, ROBIN, and SEAL. It is noteworthy that although SEAL is comparable to our watermarking against forgery attacks, the poor robustness against image distortions restricts its real-world applications. Furthermore, even though RingID gains superior performance under image distortions, the weakness against removal and forgery attacks, as demonstrated in Tables 1 and 2 restricts their real-world applications.

Ablation study of each component. To investigate the influence of each component in our ISTS, we conduct an ablation study in Table 4 to evaluate the performance without certain components. Results demonstrate that, for all removal and forgery attacks except Avg-Forgery, combining all three components achieves the best robustness. Notably, dynamic patterns are important for robustness against forgery attacks, especially for Imp-Forgery (0.72 to 0.62) and Avg-Forgery (0.60 to 0.47), because attackers can easily forge the fixed watermarking pattern even with a single image, as discussed in Section 4.1. Two-sided detection contributes greater robustness against Imp-Removal (0.71 to 0.82) as it resists attacks on opposite latent representations, which validates our findings.

Table 3. The average and worst-case detection AUC across Rotation, Noise, Blurring, Cropping, JPEG, and Diffpure. T-R means Tree-Ring, G-S means Gaussian-Shading.

AUC	T-R	RingID	G-S	ROBIN	SEAL	ISTS(Ours)
Average	0.9750	0.9874	0.9120	0.9276	0.7699	0.9742
Worst-Case	0.9276	0.9526	0.5194	0.7504	0.5228	0.9331

Table 4. Detection AUC of components in our ISTS watermarking. "w/o Dyn-Pattern", "w/o Dyn-Step" and "w/o Two-Sided" represent removing the pattern position selection, removing the injection timestep selection, and removing the two-sided detection back to one-sided, respectively.

Method	Removal(↑)			Forgery(↓)		
	Imp	Avg	VAE	Imp	Avg	VAE
ISTS	**0.8210**	**0.9900**	**0.9979**	**0.6340**	0.4737	**0.9491**
w/o Dyn-Pattern	0.7893	0.9762	0.9969	0.7202	0.6014	0.9625
w/o Dyn-Injection	0.8014	0.9809	0.9973	0.6678	**0.4440**	0.9515
w/o Two-Sided	0.7129	0.9866	0.9945	0.6412	0.4572	0.9532

6. Conclusion

In this paper, a novel content watermarking method against both removal and forgery attacks called ISTS has been proposed. ISTS utilizes instance-specific parameters to control the watermarking pattern and the injection timestep, and calculates the metric with two-sided detection. Experiments on various removal and forgery attacks have demonstrated the superior performance against existing methods.

Limitations and future works. Although our ISTS manifests the state-of-the-art performance against removal and forgery attacks, the worst-case robustness is still not satisfactory for effective removal and forgery attacks such as Imp-Removal and VAE-Forgery. As a provenance and credit attribution approach, more robust content watermarking should be developed for real-world applications.

Codes. Please find our codes and implementation details at https://github.com/hala64/ISTS.

Acknowledgement

This paper is supported by the Strategic Priority Research Program of CAS Grant XDA0480500, the Robotic AI-Scientist Platform of the Chinese Academy of Sciences, and NSFC Grants 92270001 and 12288201.

References

[1] Bang An, Mucong Ding, Tahseen Rabbani, Aakriti Agrawal, Yuancheng Xu, Chenghao Deng, Sicheng Zhu, Abdirisak Mohamed, Yuxin Wen, Tom Goldstein, et al. Benchmarking the robustness of image watermarks. ICML, 2024. 2

[2] Kasra Arabi, Benjamin Feuer, R. Teal Witter, Chinmay Hegde, and Niv Cohen. Hidden in the noise: Two-stage robust watermarking for images. In *The Thirteenth International Conference on Learning Representations*, 2025. 2

[3] Kasra Arabi, R. Teal Witter, Chinmay Hegde, and Niv Cohen. SEAL: Semantic Aware Image Watermarking, 2025. 2, 5, 6

[4] Zhongjie Ba, Yitao Zhang, Peng Cheng, Bin Gong, Xinyu Zhang, Qinglong Wang, and Kui Ren. Robust watermarks leak: Channel-aware feature extraction enables adversarial watermark manipulation. *arXiv preprint arXiv:2502.06418*, 2025. 2

[5] Ruibo Chen, Yihan Wu, Junfeng Guo, and Heng Huang. Demark: Watermark removal in large language models. *arXiv preprint arXiv:2410.13808*, 2024. 2

[6] Zhengxue Cheng, Heming Sun, Masaru Takeuchi, and Jiro Katto. Learned image compression with discretized gaussian mixture likelihoods and attention modules. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7939–7948, 2020. 2

[7] Mehdi Cherti, Romain Beaumont, Ross Wightman, Mitchell Wortsman, Gabriel Ilharco, Cade Gordon, Christoph Schuhmann, Ludwig Schmidt, and Jenia Jitsev. Reproducible scaling laws for contrastive language-image learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2818–2829, 2023. 7

[8] Miranda Christ, Sam Gunn, and Or Zamir. Undetectable watermarks for language models. In *The Thirty Seventh Annual Conference on Learning Theory*, pages 1125–1139. PMLR, 2024. 2

[9] Hai Ci, Pei Yang, Yiren Song, and Mike Zheng Shou. Ringid: Rethinking tree-ring watermarking for enhanced multi-key identification. In *European Conference on Computer Vision*, pages 338–354. Springer, 2025. 1, 2, 3, 6

[10] Ingemar Cox, Matthew Miller, Jeffrey Bloom, and Chris Honsinger. Digital watermarking. *Journal of Electronic Imaging*, 11(3):414–414, 2002. 2

[11] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021. 4, 5

[12] Ziping Dong, Chao Shuai, Zhongjie Ba, Peng Cheng, Zhan Qin, Qinglong Wang, and Kui Ren. Wmcopier: Forging invisible image watermarks on arbitrary images. *arXiv preprint arXiv:2503.22330*, 2025. 2

[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. 3

[14] 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. 2

[15] Pierre Fernandez, Hady Elsahar, I Zeki Yalniz, and Alexandre Mourachko. Video seal: Open and efficient video watermarking. *arXiv preprint arXiv:2412.09492*, 2024. 2

[16] Paul Glynn. Sony world photography award 2023: Winner refuses award after revealing AI creation. https://www.bbc.com/news/entertainment-arts-65296763, 2023. 1

[17] Sam Gunn, Xuandong Zhao, and Dawn Song. An undetectable watermark for generative image models. *arXiv preprint arXiv:2410.07369*, 2024. 1, 2

[18] Yiyang Guo, Ruizhe Li, Mude Hui, Hanzhong Guo, Chen Zhang, Chuangjian Cai, Le Wan, et al. Freqmark: Invisible image watermarking via frequency based optimization in latent space. *Advances in Neural Information Processing Systems*, 37:112237–112261, 2024. 2

[19] Hengzhi He, Peiyu Yu, Junpeng Ren, Ying Nian Wu, and Guang Cheng. Watermarking generative tabular data. *arXiv preprint arXiv:2405.14018*, 2024. 2

[20] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 3

[21] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 2

[22] Runyi Hu, Jie Zhang, Yiming Li, Jiwei Li, Qing Guo, Han Qiu, and Tianwei Zhang. Videoshield: Regulating diffusion-based video generation models via watermarking. *arXiv preprint arXiv:2501.14195*, 2025. 2

[23] Xuming Hu, Hanqian Li, Jungang Li, Yu Huang, and Aiwei Liu. Videomark: A distortion-free robust watermarking framework for video diffusion models. *arXiv preprint arXiv:2504.16359*, 2025. 2

[24] Yuepeng Hu, Zhengyuan Jiang, Moyang Guo, and Neil Zhenqiang Gong. A transfer attack to image watermarks. *arXiv preprint arXiv:2403.15365*, 2024. 2

[25] Huayang Huang, Yu Wu, and Qian Wang. Robin: Robust and invisible watermarks for diffusion models with adversarial optimization. *Advances in Neural Information Processing Systems*, 37:3937–3963, 2024. 2, 3, 6

[26] Anubhav Jain, Yuya Kobayashi, Naoki Murata, Yuhta Takida, Takashi Shibuya, Yuki Mitsufuji, Niv Cohen, Nasir Memon, and Julian Togelius. Forging and Removing Latent-Noise Diffusion Watermarks Using a Single Image, 2025. 1, 2, 4, 5, 6, 13

[27] Zhengyuan Jiang, Jinghuai Zhang, and Neil Zhenqiang Gong. Evading watermark based detection of ai-generated

content. In *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security*, pages 1168–1181, 2023. 2

[28] Nikola Jovanović, Robin Staab, and Martin Vechev. Watermark stealing in large language models. *arXiv preprint arXiv:2402.19361*, 2024. 2

[29] Andre Kassis and Urs Hengartner. Unmarker: a universal attack on defensive image watermarking. In *2025 IEEE Symposium on Security and Privacy (SP)*, pages 2602–2620. IEEE, 2025. 1

[30] John Kirchenbauer, Jonas Geiping, Yuxin Wen, Jonathan Katz, Ian Miers, and Tom Goldstein. A watermark for large language models. In *International Conference on Machine Learning*, pages 17061–17084. PMLR, 2023. 2

[31] John Kirchenbauer, Jonas Geiping, Yuxin Wen, Manli Shu, Khalid Saifullah, Kezhi Kong, Kasun Fernando, Aniruddha Saha, Micah Goldblum, and Tom Goldstein. On the reliability of watermarks for large language models. *arXiv preprint arXiv:2306.04634*, 2023. 2

[32] Rohith Kuditipudi, John Thickstun, Tatsunori Hashimoto, and Percy Liang. Robust distortion-free watermarks for language models. *arXiv preprint arXiv:2307.15593*, 2023. 2

[33] Martin Kutter, Sviatoslav V Voloshynovskiy, and Alexander Herrigel. Watermark copy attack. In *Security and Watermarking of Multimedia Contents II*, pages 371–380. SPIE, 2000. 2

[34] Wenda Li, Huijie Zhang, and Qing Qu. Shallow diffuse: Robust and invisible watermarking through low-dimensional subspaces in diffusion models. *arXiv preprint arXiv:2410.21088*, 2024. 2, 3, 6

[35] Xiang Li, Chan Lu, Danni Cheng, Wei-Hong Li, Mei Cao, Bo Liu, Jiechao Ma, and Wei-Shi Zheng. Towards photo-realistic visible watermark removal with conditional generative adversarial networks. In *International conference on image and graphics*, pages 345–356. Springer, 2019. 2

[36] Gloria Liu. The World's Smartest Artificial Intelligence Just Made Its First Magazine Cover. https://www.cosmopolitan.com/lifestyle/a40314356/dall-e-2-artificial-intelligence-cover/, 2022. 1

[37] Yepeng Liu and Yuheng Bu. Adaptive text watermark for large language models. *arXiv preprint arXiv:2401.13927*, 2024. 2

[38] 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. *arXiv preprint arXiv:2410.05470*, 2024. 2

[39] Yixin Liu, Kai Zhang, Yuan Li, Zhiling Yan, Chujie Gao, Ruoxi Chen, Zhengqing Yuan, Yue Huang, Hanchi Sun, Jianfeng Gao, Lifang He, and Lichao Sun. Sora: A Review on Background, Technology, Limitations, and Opportunities of Large Vision Models, 2024. 1

[40] Nils Lukas, Abdulrahman Diaa, Lucas Fenaux, and Florian Kerschbaum. Leveraging optimization for adaptive attacks on image watermarks. *arXiv preprint arXiv:2309.16952*, 2023. 2

[41] Andreas Müller, Denis Lukovnikov, Jonas Thietke, Asja Fischer, and Erwin Quiring. Black-box forgery attacks on semantic watermarks for diffusion models. *arXiv preprint arXiv:2412.03283*, 2024. 1, 2, 4, 5, 6, 12, 13

[42] Weili Nie, Brandon Guo, Yujia Huang, Chaowei Xiao, Arash Vahdat, and Anima Anandkumar. Diffusion models for adversarial purification. *arXiv preprint arXiv:2205.07460*, 2022. 2

[43] Mikhail Pautov, Danil Ivanov, Andrey V Galichin, Oleg Rogov, and Ivan Oseledets. Spread them apart: Towards robust watermarking of generated content. *arXiv preprint arXiv:2502.07845*, 2025. 2

[44] 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. 12

[45] 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. 7

[46] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-Shot Text-to-Image Generation, 2021. 1

[47] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2021. 1

[48] 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. 3, 5

[49] Mehrdad Saberi, Vinu Sankar Sadasivan, Keivan Rezaei, Aounon Kumar, Atoosa Chegini, Wenxiao Wang, and Soheil Feizi. Robustness of ai-image detectors: Fundamental limits and practical attacks. *arXiv preprint arXiv:2310.00076*, 2023. 2

[50] Gustavo Santana. Stable-diffusion-prompts, 2024. 5

[51] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 2

[52] 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. 2

[53] Chen Sun, Haiyang Sun, Zhiqing Guo, Yunfeng Diao, Liejun Wang, Dan Ma, Gaobo Yang, and Keqin Li. Diffmark: Diffusion-based robust watermark against deepfakes. *arXiv preprint arXiv:2507.01428*, 2025. 2

[54] 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. 2

[55] Sviatoslav Voloshynovskiy, Shelby Pereira, Thierry Pun, Joachim J Eggers, and Jonathan K Su. Attacks on digital watermarks: classification, estimation based attacks, and benchmarks. *IEEE communications Magazine*, 39(8):118–126, 2001. 2

[56] Yaopeng Wang, Huiyu Xu, Zhibo Wang, Jiacheng Du, Zhichao Li, Yiming Li, Qiu Wang, and Kui Ren. PT-Mark: Invisible Watermarking for Text-to-image Diffusion Models via Semantic-aware Pivotal Tuning, 2025. 1, 2

[57] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 7

[58] Yuxin Wen, John Kirchenbauer, Jonas Geiping, and Tom Goldstein. Tree-rings watermarks: Invisible fingerprints for diffusion images. In *Thirty-Seventh Conference on Neural Information Processing Systems*, 2023. 1, 2, 3, 5, 6, 7, 12

[59] Yangxinyu Xie, Xiang Li, Tanwi Mallick, Weijie Su, and Ruixun Zhang. Debiasing watermarks for large language models via maximal coupling. *Journal of the American Statistical Association*, (just-accepted):1–21, 2025. 2

[60] Pei Yang, Hai Ci, Yiren Song, and Mike Zheng Shou. Can simple averaging defeat modern watermarks? In *The Thirty-Eighth Annual Conference on Neural Information Processing Systems*, 2024. 1, 2, 4, 5, 6, 13

[61] Zijin Yang, Kai Zeng, Kejiang Chen, Han Fang, Weiming Zhang, and Nenghai Yu. Gaussian shading: Provable performance-lossless image watermarking for diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12162–12171, 2024. 1, 2, 3, 6

[62] Zijin Yang, Xin Zhang, Kejiang Chen, Kai Zeng, Qiyi Yao, Han Fang, Weiming Zhang, and Nenghai Yu. Gaussian shading++: Rethinking the realistic deployment challenge of performance-lossless image watermark for diffusion models. *arXiv preprint arXiv:2504.15026*, 2025. 2

[63] Jongmin Yoon, Sung Ju Hwang, and Juho Lee. Adversarial purification with score-based generative models. In *International Conference on Machine Learning*, pages 12062–12072. PMLR, 2021. 2

[64] Kevin Alex Zhang, Lei Xu, Alfredo Cuesta-Infante, and Kalyan Veeramachaneni. Robust invisible video watermarking with attention. *arXiv preprint arXiv:1909.01285*, 2019. 2

[65] Lu Zhang and Liang Zeng. Sat-ldm: Provably generalizable image watermarking for latent diffusion models with self-augmented training. *arXiv preprint arXiv:2501.00463*, 2024. 2

[66] Lijun Zhang, Xiao Liu, Antoni Viros i Martin, Cindy Xiong Bearfield, Yuriy Brun, and Hui Guan. Attack-resilient image watermarking using stable diffusion. In *The Thirty-Eighth Annual Conference on Neural Information Processing Systems*, 2024. 1, 2, 3, 6

[67] 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. 7

[68] Zhaoxi Zhang, Xiaomei Zhang, Yanjun Zhang, Leo Yu Zhang, Chao Chen, Shengshan Hu, Asif Gill, and Shirui Pan. Large language model watermark stealing with mixed integer programming. *arXiv preprint arXiv:2405.19677*, 2024. 2

[69] Xuandong Zhao, Prabhanjan Ananth, Lei Li, and Yu-Xiang Wang. Provable robust watermarking for ai-generated text. *arXiv preprint arXiv:2306.17439*, 2023. 2

[70] 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. *Advances in neural information processing systems*, 37:8643–8672, 2024. 8

[71] 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. 2

[72] Yihao Zheng, Haocheng Xia, Junyuan Pang, Jinfei Liu, Kui Ren, Lingyang Chu, Yang Cao, and Li Xiong. Tabularmark: Watermarking tabular datasets for machine learning. In *Proceedings of the 2024 on ACM SIGSAC conference on computer and communications security*, pages 3570–3584, 2024. 2

[73] Yifan Zhu, Lijia Yu, and Xiao-Shan Gao. Provable watermarking for data poisoning attacks. *arXiv preprint arXiv:2510.09210*, 2025. 2

A. Appendix of Paper "Towards Robust Content Watermarking Against Removal and Forgery Attacks"

A.1. Experimental Details

Watermarking hyperparameter assignment. After gaining the label $y^c \in [C]$ for C-classes clustering, we need to assign corresponding injection timestep t and pattern location offset l. Specifically, we denote the injection timestep range in $[T_1, T_2]$, the pattern location offset $l = (l_x, l_y)$ range in $[l_x^1, l_x^2] \times [l_y^1, l_y^2]$. The injection timestep t is set as

$$t = T_1 + [y^c \bmod (T_2 - T_1)],$$

the pattern position offset is set as

$$l_x = l_x^1 + \left[y^c \bmod (l_x^2 - l_x^1)\right], l_y = l_y^1 + \left[\left(y^c // (l_y^2 - l_y^1)\right) \bmod (l_y^2 - l_y^1)\right].$$

The watermarking pattern after location offset should be

$$\text{Offset}(W, l)_{i,j} = W_{i+l_x, j+l_y}.$$

In this paper, we set the clustering class $C = 1024$, the timestep range in $[10, 20]$, and the pattern position offset be $[-12, 12]^2$ by default.
Watermark injection. After getting the watermarking pattern $W^o = \text{Offset}(W, l)$, we add the watermarking pattern into the t-step latent z_t. Building upon with the method proposed by Wen et al. [58], denote the watermarking mask in the frequency domain be M, M is also deployed with location offset l similar as W, denoted as

$$M^o_{i,j} = M_{i+l_x, j+l_y}.$$

The watermarking channel is 0 and the radius is set to be 20. Then we replace the corresponding pixel $k = (i_k, j_k)$ of z_t by W^o under mask M^o in the frequency domain, i.e.

$$\mathcal{F}(z_t)_k = \begin{cases} W^o_k, & \text{if } k \in M^o, \\ \mathcal{F}(\mathcal{M}_{T \to t}(p, z_T))_k, & \text{otherwise.} \end{cases}$$

Finally, the watermarked t-step latent z_t^w is obtained by inverse Fourier transformation:

$$z_t^w = \mathcal{F}^{-1}(\mathcal{F}(z_t)),$$

after the modification on $\mathcal{F}(z_t)$.
To make the notation simple, in our Algorithm 2, we donate the above injection process as

$$z_t^w = z_t \oplus \text{Offset}(W, l).$$

Parameter selector. For a new prompt p_{new} and the corresponding generated image I_{new}, we need to assign its watermarking parameter with our parameter selector $f = h \circ g$, where g is a pre-trained CLIP feature extractor, and f is the classifier trained by existing features and their assigned labels.
To reduce the computational complexity, we use a very simple two-layer neural network to align extracted features with their watermarking parameters, where the hidden dimension is same as the input feature dimension, and apply ReLU activation, Batch Normalization with $p = 0.5$ dropout rate.

A.2. Details on Removal and Forgery Attacks

Imp-Removal. In Müller et al. [41], they try to remove watermarks by optimizing the follow equation:

$$\mathcal{L}(\delta) = \|\mathcal{M}_{0 \to T}(p, z_0^w + \delta) + z_T^w\|_2,$$

where $\mathcal{M}$ is the surrogate diffusion model, p is the text prompt, $z_0^w = \mathcal{E}(x^w)$ derive from the obtained watermarked image x^w with encoder $\mathcal{E}$, $z_T^w = \mathcal{M}_{0 \to T}(p, z_0^w)$ is an estimation of T-step latent after DDIM inversion from z_0^w. Tne Imp-Removal attack aims to induce watermarked latent to the opposite position in timestep T, which could be vulnerable to our two-sided detection. Finally, the image after Imp-Removal attack $\hat{x}^w$ is decoded by decoder $\mathcal{D}$ from the perturbed latent:

$$\hat{x}^w = \mathcal{D}(z_0^w + \delta).$$

Following the setting provided in Müller et al. [41], we set the optimization steps be 150, with learning rate be 0.01. Furthermore, it is noteworthy that, Müller et al. [41] considers the surrogate model scenario, where the attack model be Stable-Diffusion 2.1 (SD 2.1), and the victim models be different, like SD 2.1-Anime (SD 2.1 Finetuned on Anime), Stable Diffusion XL [44], etc. In this paper, we consider

a stronger case for attackers, that they can use the same diffusion model as surrogate, inducing a white-box rather than grey-box scenario. As our evaluation based on this white-box scenario, the robustness of our watermarking will be not bad than those from grey-box scenario.

Imp-Forgery. In Müller et al. [41], they want to forge existing watermarking method by optimizing the following loss:

$$\mathcal{L}(\delta) = \|\mathcal{M}_{0\to T}(p, z_0^c + \delta) - z_T^w\|_2,$$

where $z_0^c = \mathcal{E}(x^c)$ derive from the clean image x^c ready to be forged, $z_T^w = \mathcal{M}_{0\to T}(p, z_0^w)$ is an estimation of T-step latent after DDIM inversion from z_0^w, and $z_0^w = \mathcal{E}(x^w)$ derive from the obtained watermarked image x^w. Other notations are similar to Imp-Removal. Similarly, the optimization steps are 150 with learning rate is 0.01 following the setting of Müller et al. [41]. We also evaluate on the white-box scenario in Imp-Forgery attack. The image after Imp-Forgery attack $\hat{x}^w$ is decoded by decoder $\mathcal{D}$ from the perturbed latent:

$$\hat{x}^w = \mathcal{D}(z_0^c + \delta).$$

Avg-Removal and Avg-Forgery. In Yang et al. [60], they find that watermarking patterns can be revealed by averaging a collection of watermarked and non-watermarked images. Specifically, they propose averaging over N images and calculating their residual as:

$$\delta = \frac{1}{N}\left(\sum_{i=1}^{N} x_{w,i} - \sum_{i=1}^{N} x_{c,i}\right),$$

where $\{x_{c,i}\}_{i\in[N]}$ are clean non-watermarked images, $\{x_{w,i}\}_{i\in[N]}$ are watermarked images.
In Avg-Removal attack, they modify the watermarked image x^w to $\hat{x}^w$ by

$$\hat{x}^w = x^w - \delta,$$

where in Avg-Forgery attacks, they modify the clean image x^c to $\hat{x}^w$ by

$$\hat{x}^w = x^c + \delta.$$

In this paper, we first generate 100 pairs of non-watermarked and watermarked images to extract the residual δ, then conduct Avg-Removal and Avg-Forgery attacks for each watermarked/non-watermarked image.

VAE-Removal. In Jain et al. [26], similar to Müller et al. [41], they try to remove and forge watermarks with a single watermarked image. Unlike Müller et al. [41] uses a surrogate diffusion model to optimize perturbations in the T-step noise latent space, Jain et al. [26] directly optimize attacks on latent image space with only the surrogate VAE encoder. Specifically, in VAE-Removal attack, they optimize injected noise δ by:

$$\min_{\delta} \|\mathcal{E}(x^w + \delta) - \mathcal{E}(\mu_{x^w})\|_2 + \lambda\|\delta\|_2,$$

where x^w is the watermarked image ready to be removed, $\mathcal{E}$ is the VAE encoder, λ is the balancing hyperparameter, and μ_{x^w} is the plain image with all values equal to the mean of the watermarked image x^w.
In this paper, followed the setting of Jain et al. [26], we choose the VAE encoder from Stable Diffusion v1.4, and set λ be 5×10^4.

VAE-Forgery. Similar to VAE-Removal attacks, VAE-Forgery proposed by Jain et al. [26] also optimize the perturbation at the latent image space. Specifically, VAE-Forgery works as:

$$\min_{\delta} \|\mathcal{E}(x^c + \delta) - \mathcal{E}(x^w)\|_2 + \lambda\|\delta\|_2,$$

where x^c is the clean non-watermarked image ready to be forged, x^w is the obtained watermarked image, other notations are similar to VAE-Removal.
Same as VAE-Removal, the VAE encoder is from Stable Diffusion v1.4, and the hyperparameter $\lambda = 5 \times 10^4$.

A.3. Different Intermediate Steps for Watermarking Injection

To further investigate the impact of intermediate steps for watermarking injection, we evaluate different ranges of injection steps from $t = 5$ to $t = 45$, the overall diffusion steps is $T = 50$. As our watermarking injection dynamically select $t \in [10, 20]$, to ensure the consistency, we evaluate different steps with the same range of variation, from $[5, 15]$ to $[35, 45]$. We use Imp-Removal and Imp-Forgery as the removal and forgery attacks, results are shown in Figure 5.
It reveals that although the detection AUC remains sufficiently high for original watermarking without attacks across different intermediate steps, the performance under removal and forgery attacks have changed considerably. As steps go larger, Detection AUC becomes worse for both removal and forgery attacks, implying a relative small injection steps when designing robust watermarking.

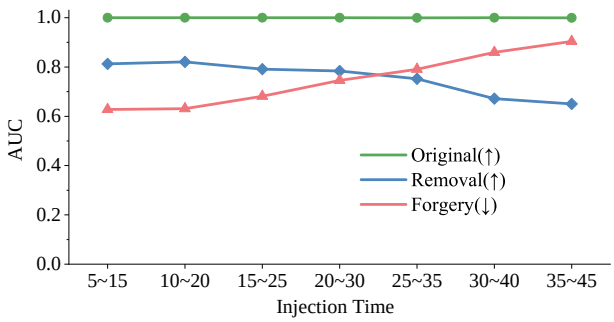

Figure 5. Different injection steps.

A.4. Potential Side-Channel Attacks on ISTS

A simple side-channel attack of our ISTS watermarking is the potential leakage of the public CLIP extractors and even the parameter selector model. However, we believe that information extracted from CLIP features or the parameter selector model of potentially leaked watermarked images does not compromise the security of our ISTS watermarking scheme. This is because the assigned labels y_p^c can be obfuscated using a (pseudo-)random permutation $\mathcal{R}$ controlled by a secret key. Given a large number of clusters (e.g., 1024), it becomes infeasible for an attacker to recover the injection parameters (t, l), as they cannot infer the permuted label $\mathcal{R}(y_p^c)$ because 2^{1024} is a infeasible large number for any attackers. Moreover, in practical deployments, watermarking methods are typically applied to proprietary models, where adversaries are limited to black-box access via APIs and do not have visibility into the model's internal structures or algorithms.