# OptMark: Robust Multi-bit Diffusion Watermarking via Inference Time Optimization

**Jiazheng Xing** [1, 2*], **Hai Ci** [2*], **Hongbin Xu** [2], **Hangjie Yuan** [1], **Yong Liu** [1†], **Mike Zheng Shou** [2]

[1]Zhejiang University
[2] Show Lab, National University of Singapore
jiazhengxing@zju.edu.cn, yongliu@iipc.zju.edu.cn
{cihai03, mike.zheng.shou}@gmail.com

## Abstract

Watermarking diffusion-generated images is crucial for copyright protection and user tracking. However, current diffusion watermarking methods face significant limitations: zero-bit watermarking systems lack the capacity for large-scale user tracking, while multi-bit methods are highly sensitive to certain image transformations or generative attacks, resulting in a lack of comprehensive robustness. In this paper, we propose **OptMark**, an optimization-based approach that embeds a robust multi-bit watermark into the intermediate latents of the diffusion denoising process. OptMark strategically inserts a structural watermark early to resist generative attacks and a detail watermark late to withstand image transformations, with tailored regularization terms to preserve image quality and ensure imperceptibility. To address the challenge of memory consumption growing linearly with the number of denoising steps during optimization, OptMark incorporates adjoint gradient methods, reducing memory usage from O(N) to O(1). Experimental results demonstrate that OptMark achieves invisible multi-bit watermarking while ensuring robust resilience against valuemetric transformations, geometric transformations, editing, and regeneration attacks.

## 1 Introduction

In the AIGC era, diffusion models (Ho, Jain, and Abbeel 2020; Song, Meng, and Ermon 2020; Rombach et al. 2022) have become a cornerstone of digital content creation, enabling the generation of hyper-realistic images. This advancement revolutionizes visual content production while raising critical intellectual property and content safety challenges in the digital age. As a crucial copyright protection technology, invisible watermarking enables AIGC service providers to embed imperceptible identifiers into generated content, facilitating traceability and ownership verification. This paper explores multi-bit invisible watermarking for diffusion-generated content, focusing on copyright protection and traceability.

Current watermarking approaches fall into two camps: pixel-level and semantic-level. Pixel-level watermarking methods, such as HiDDeN (Zhu 2018), SSL (Fernandez et al. 2022), WAM (Sander et al. 2024), and Stable Signature (Fernandez et al. 2023), embed watermarks directly at the pixel level. While these methods are straightforward to implement, they exhibit limited robustness against regeneration attacks (Zhao et al. 2023). Semantic-level watermarking methods usually embed watermarks during the image generation process and alter the semantic layout of the generated images. A typical approach is to embed handcrafted watermark patterns in the diffusion noise. Compared with pixel-level methods, these approaches are more robust to regeneration attacks, yet they remain vulnerable to certain image transformations and often lack sufficient capacity to embed more bits. Specifically, Tree-Ring (Wen et al. 2023) is susceptible to cropping and scaling, while Gaussian Shading (Yang et al. 2024) is vulnerable to geometric attacks that disrupt the order of patches, such as horizontal flipping. Furthermore, methods such as RingID (Ci et al. 2024b) and WIND (Arabi et al. 2024) lack sufficient capacity to embed adequate watermark bits, limiting their scalability. Overall, significant challenges remain in balancing robustness and capacity in existing approaches.

In this paper, we propose **OptMark**, a novel semantic-level multi-bit watermarking approach that ensures ample capacity while achieving comprehensive robustness against four common types of attacks: valuemetric, geometric, editing, and regeneration, as shown in Fig. 1. To achieve this, OptMark optimizes the watermarks in an end-to-end manner during the diffusion inference process. Unlike prior works that rely on handcrafted watermark patterns (Wen et al. 2023; Ci et al. 2024b; Yang et al. 2024), our approach offers two key advantages through end-to-end learning: 1) *Enhanced robustness*: By seamlessly integrating with diverse training-time image augmentations, OptMark improves resilience against a wide range of attacks, whereas manually designed watermarks struggle to cover all possible scenarios. 2) *Greater flexibility*: End-to-end optimization allows for the efficient embedding of a larger number of bits, as the process is fully automated.

To establish this end-to-end optimization framework with comprehensive robustness, high image quality, and low GPU memory overhead, we introduce three key designs: 1) **Comprehensive Robustness**: We adopt a dual watermarking mechanism, optimizing a structure watermark in the initial diffusion noise to resist generative attacks and a detail wa-

*These authors contributed equally.
†Corresponding author.

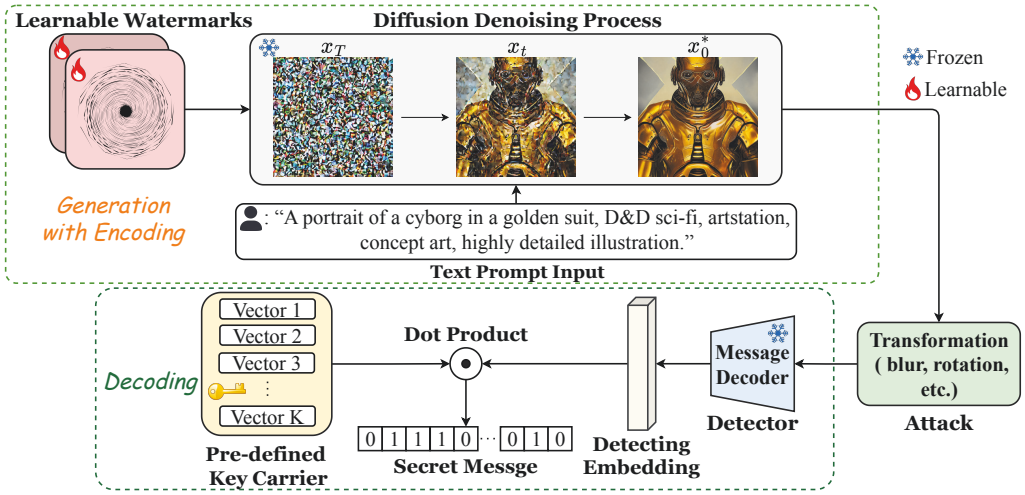


Figure 1: Pipeline of our end-to-end optimized **OptMark**. During generation, the robust watermark is injected into the diffusion latent space via inference-time optimization. In decoding, a pre-trained message decoder extracts the watermark embedding, which is matched against a predefined key carrier to recover the secret message.

termark in one late denoising step to counter image transformations. 2) **Minimal Impact on Image Quality**: We develop specialized embedding strategies and constraints to regulate the shape and statistical properties of the learned watermarks, ensuring high image quality and imperceptibility. 3) **Efficient GPU Memory Usage**: To reduce GPU memory overhead, we introduce the adjoint method for computing gradients on learnable watermarks, lowering memory consumption from $O(N)$ to $O(1)$. Extensive experiments demonstrate that our method significantly outperforms state-of-the-art approaches in robustness, with sufficient bit capacity and high generated image quality.

## 2 Related Work

### 2.1 Pixel-Level Watermark

Pixel-level watermarking typically embeds invisible watermarks directly into the image pixel domain. Mainstream approaches can be categorized into two types: optimization-based methods and encoder-decoder methods. Representative optimization-based approaches, such as FNNS (Kishore et al. 2021) and SSL (Fernandez et al. 2022), iteratively optimize a small perturbation on the cover image so that the image features extracted by a pre-trained model can reliably recover the target watermark bits. In contrast, encoder-decoder methods (Zhu 2018; Tancik, Mildenhall, and Ng 2020; Fernandez et al. 2023; Ci et al. 2024a; Sander et al. 2024) train watermark encoders and decoders on a large set of images with different watermark bit sequences, enabling on-the-fly embedding of watermark bits into images. While pixel-level watermarking is imperceptible to the human eye, it has been shown to be inherently vulnerable to regeneration attacks (Zhao et al. 2023).

### 2.2 Semantic-Level Watermark

Semantic-level watermark approaches embed watermarks during the diffusion generation process, altering the semantic content and layout of the generated image, and improving robustness against regeneration attacks. Some methods train diffusion plugins (Feng et al. 2024; Min et al. 2024) for semantic watermarking, but they require expensive training and struggle to achieve optimal robustness. While Tree-Ring (Wen et al. 2023) pioneered another direction by injecting a handcrafted tree-ring pattern into the initial diffusion noise as a zero-bit watermark. Subsequent works (Ci et al. 2024b; Yang et al. 2024; Zhang et al. 2024; Huang, Wu, and Wang 2024; Gunn, Zhao, and Song 2024) have improved its robustness or imperceptibility. However, they either remain vulnerable to geometric attacks (Yang et al. 2024) or lack the capacity to embed sufficient multi-bit information (Ci et al. 2024b; Zhang et al. 2024; Huang, Wu, and Wang 2024). Our proposed method, OptMark, belongs to the semantic-level watermarking. It is the first approach to achieve both sufficient multi-bit capacity and comprehensive robustness against common image transformations and generative attacks.

## 3 Method

### 3.1 Preliminary

**Diffusion Models.** Diffusion models (Ho, Jain, and Abbeel 2020; Song, Meng, and Ermon 2020) progressively convert standard Gaussian noise $x_T \sim \mathcal{N}(0, \mathbf{I})$ into samples from the true data distribution $x_0 \sim q(x)$ over $T$ reverse (denoising) steps. The forward (noising) process is defined as:

$$q(x_t|x_{t-1}) = \mathcal{N}(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t\mathbf{I}), \quad (1)$$

where $\{\beta_t\}_{t=1}^T \in (0,1)$ is the scheduled variance, and $x_t$ can be sampled directly from $x_0$ as:

$$x_t = \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon, \quad (2)$$

where $\bar{\alpha}_t = \prod_{i=0}^{t}(1-\beta_t)$ and $\epsilon \sim \mathcal{N}(0, \mathbf{I})$. Subsequently, a network $\epsilon_\theta$ is trained to predict the added noise at each

step, with the following objective:

$$\mathbb{E}_{x_0,t\sim\texttt{Uniform}(1,T),\epsilon\in\mathcal{N}(0,\mathbf{I})}\left[\|\epsilon-\epsilon_\theta\left(x_t,t,\psi(p)\right)\|_2^2\right], \quad (3)$$

where $x_t$ represents the noisy latent at timesteps $t$ and $\psi(p)$ denotes the embedding of the text input prompt $p$. The reverse (generation) process can be written as:

$$\begin{aligned} x_{t-1} =& \sqrt{\alpha_{t-1}}\left(\frac{x_t-\sqrt{1-\alpha_t}\epsilon_\theta\left(x_t\right)}{\sqrt{\alpha_t}}\right) \\ &+\sqrt{1-\alpha_{t-1}-\sigma_t^2}\cdot\epsilon_\theta\left(x_t\right)+\sigma_t\epsilon_t. \end{aligned} \quad (4)$$

When $\sigma_t = 0$, it is a DDIM sampler (Song, Meng, and Ermon 2020). When $\sigma_t = \sqrt{\left(1-\alpha_{t-1}\right)/\left(1-\alpha_t\right)}\sqrt{1-\alpha_t/\alpha_{t-1}}$, it is a DDPM sampler (Ho, Jain, and Abbeel 2020).

**Background and Task Definition.** In the multi-bit watermarking scenario for diffusion models, OptMark embeds a $k$-bit invisible watermark message $m$ into the generation process to produce a watermarked image $x_0^*$. When these images are disseminated online, they may undergo various attacks $\mathcal{T}$. For copyright verification or user identification, the model owner decodes the potentially distorted image $\mathcal{T}(x_0^*)$ to recover $\hat{m}$ and compares it to the original watermark $m$.

## 3.2 Overview

Figure 1 illustrates the OptMark's end-to-end pipeline, which comprises two stages: *Watermark Encoding* and *Decoding*. In the *Watermark Encoding* stage, learnable watermark vectors are injected into the diffusion latents during inference to produce a watermarked image $x_0^*$. An inference-time optimization strategy balances watermark robustness against visual fidelity. In the *Decoding* stage, we employ a pre-trained, self-supervised image encoder (Caron et al. 2021) as the message decoder to extract the embedded watermark representation from versions of $x_0^*$ subjected to attacks $\mathcal{T}$. Finally, the $k$-bit message is recovered by computing the dot product between this representation and a predefined set of $k$ carrier vectors.

## 3.3 Dual-Watermark for Diffusion Models

**Watermark Encoding** Compared with recent pixel-level watermarking methods (Kishore et al. 2021; Fernandez et al. 2022), which exhibit poor robustness against regeneration attacks (Zhao et al. 2023), OptMark embeds messages directly into the diffusion denoising process and thus achieves significantly higher resistance to these attacks. The diffusion model's denoising trajectory can be divided into two stages: *structure formulation* and *detail refinement*. We therefore propose injecting different watermarks at each stage, with each targeting a distinct semantic level, to enhance robustness against a wide range of attacks. However, since imprinting the watermark into the denoising process is an increasing entropy reaction, excessive introduction of the watermark can negatively impact the quality of image generation. To balance the watermark robustness and image quality, OptMark inserts exactly one watermark per stage: a *structure watermark* during the first stage, injected into high-level semantic features to leave a persistent mark that is difficult to erase through generative attacks; and a *detail watermark* during the second stage embedded at a finer, near-pixel level to withstand geometric and volumetric attacks while accelerating convergence.

We consider a standard diffusion framework using the DDIM sampler (Song, Meng, and Ermon 2020). Fig. 2 depicts the watermark embedding process in OptMark. Given standard Gaussian initial noise $x_T \sim \mathcal{N}\left(0,\mathbf{I}\right)$, the model predicts the noise $\epsilon_\theta$ at each denoising timestep $t$ via:

$$\hat{\epsilon}_t = \begin{cases} \epsilon_\theta\left(\mathcal{F}_s\left(x_t,w_s\right),t,\psi\left(p\right)\right) & \text{if } t=t_s, \\ \epsilon_\theta\left(\mathcal{F}_d\left(x_t,w_d\right),t,\psi(p)\right) & \text{if } t=t_d, \\ \epsilon_\theta(x_t,t,\psi(p)) & \text{otherwise}, \end{cases} \quad (5)$$

where $w_s$ and $w_d$ represent the structure and detail watermark, respectively, both initialized with a Gaussian distribution. $\mathcal{F}_s$ and $\mathcal{F}_d$ specify the corresponding watermark-embedding operator.

**Choices of Watermark Position** We inject the structure watermark $w_s$ at the initial timestep $t_s = T$ for two reasons: (i) injecting at initialization enhances robustness against generative attacks; and (ii) the latents $x_T$ follow the standard normal distribution $\mathcal{N}\left(0,\mathbf{I}\right)$, which serves as a reference to constrain the post-embedding distribution and thus minimize any degradation in generation quality.

For the detail watermark $w_d$, we need to select an appropriate timestep $t_d$ after the semantic generation process, ensuring that the introduction of $w_d$ does not distort the semantics of the generated image. At the same time, this step should not be too close to the pixel level, as pixel-level watermarks are more vulnerable to regeneration attacks and prone to introducing visible artifacts. Fig. 3 shows the evolution of the mean values of classifier-free guidance noise throughout the generation process: $s\cdot(Condition - Uncondition)$, where "Condition" and "Uncondition" represent the predicted noise with and without text conditioning, and $s$ is the guidance scale. We can observe that over timesteps 0 to 400, the variation in guidance noise decreases significantly, indicating that the fundamental semantics have been established. Based on the ablation study detailed in the Appendix Sec. B.2, we set $t_d \in [200, 300]$ to balance watermark robustness and image quality.

**Watermark Decoding** Following SSL (Fernandez et al. 2022), we employ a pre-trained image feature extractor $\mathcal{D}_{msg}$ (e.g., DINO (Caron et al. 2021)) as our message decoder. Given a watermarked image $x_0^*$, we compute its embedding $E_w = \mathcal{D}_{msg}\left(x_0^*\right) \in \mathbb{R}^{1\times D}$, and denote the secret $k$-bit message as $m = (m_1,\ldots,m_k) \in \{-1,1\}^k$. We predefine a set of carrier vectors $\{a_i\}_{i=1}^k, a_i \in \mathbb{R}^D$, each initialized by whitening on a large natural-image dataset to ensure that decoding on arbitrary (non-watermarked) images yields i.i.d. Bernoulli(0.5) bits. The recovered message is then:

$$\hat{m} = \left[sign\left(E_w\cdot a_1^\top\right),\cdots,sign\left(E_w\cdot a_k^\top\right)\right], \quad (6)$$

During training, the watermark decoding loss is defined as the hinge loss with margin $\mu \geq 0$ on the projections:

$$\mathcal{L}_{msg} = \frac{1}{k}\sum_{i=1}^{k} max(0,(\mu-(E_w\cdot a_i^\top)\cdot m_i)). \quad (7)$$

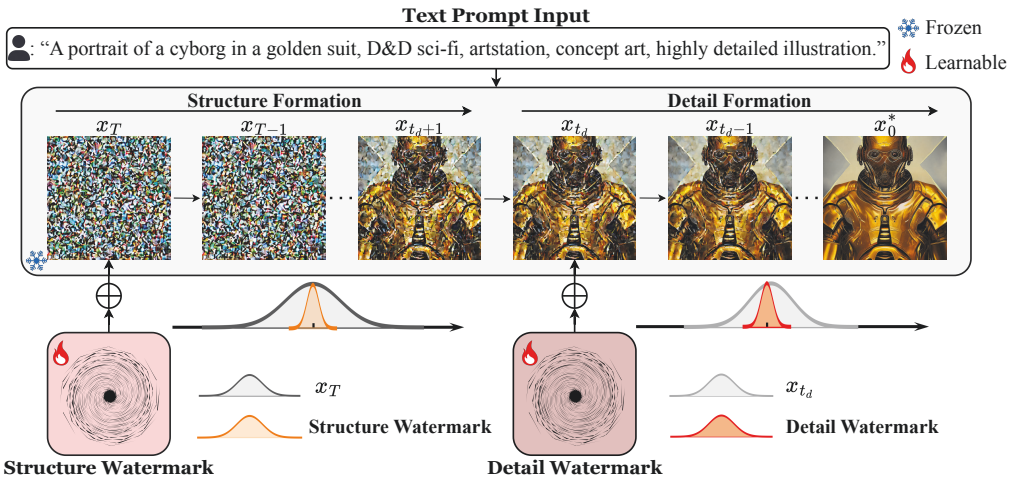


Figure 2: OptMark's imprinting process consists of two sequential stages: first, a structure watermark is injected into the initial latent state of generation; then, a detail watermark is embedded at an intermediate timestep. These complementary watermarks work in concert to maximize overall robustness.

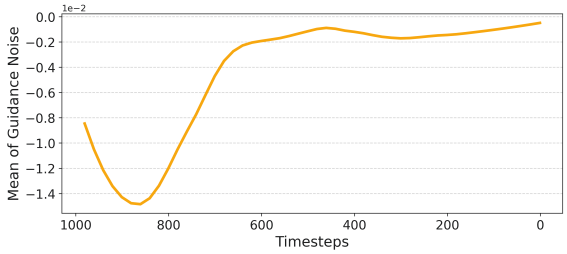


Figure 3: Predicted Guidance Noise during generation.

### 3.4 Balancing Robustness and Image Quality

**Quality-Preserving Components** To minimize watermarking's impact on visual fidelity, we propose three complementary components: *watermark initialization*, *embedding strategy*, and *regularization loss*. Our optimization targets two criteria: (i) the latent distribution before and after watermark embedding remains as close as possible; and (ii) the embedded watermark follows a low-variance Gaussian profile, as diffusion models are well trained to handle small Gaussian perturbations.

Based on the above design principles, we initialize both the structure watermark and detail watermark as $w_s^{init}, w_d^{init} \sim \mathcal{N}(0, 0.01)$. For the structure watermark, since it is embedded into the initial diffusion latent $x_T \sim \mathcal{N}(0, \mathbf{I})$, we apply a two-step normalization within the embedding operator $\mathcal{F}_s$ to preserve unit variance:

$$x_T^w = w_s + \sqrt{\frac{\text{var}(x_T) - \text{var}(w_s)}{\text{var}(x_T)}} \cdot x_T, \tag{8}$$

$$x_T^w = \sqrt{\frac{\text{var}(x_T)}{\text{var}(x_T^w)}} \cdot x_T^w, \tag{9}$$

where var$(\cdot)$ indicates the variance of data. The derivation and proof can be found in the Appendix Sec. A. Additionally, we impose an L2 regularization to ensure that the mean of the watermarked initial diffusion latent remains nearly unchanged to its original value before watermarking:

$$\mathcal{L}_{init} = \mathcal{L}_{mean}(x_T^w, x_T) = (\text{mean}(x_T^w) - \text{mean}(x_T))^2, \tag{10}$$

where mean$(\cdot)$ indicates the mean of data.

For the detail watermark, we also aim to minimize the impact of the embedding operator $\mathcal{F}_d$ on the DDIM sampling. By Eq. 4, the reverse process is robust to small Gaussian perturbations $\sigma_t \epsilon_t$. Thus, at $t = t_d$ we replace the term $\sigma_{t_d}\epsilon_{t_d}$ with the detail watermark $w_d \sim \mathcal{N}(0, 0.01)$, initializing $\sigma_{t_d} = 0.1$; for all other timesteps we use $\sigma_t = 0$.

In addition, we further introduce losses to separately constrain the watermarks' low-order statistics (mean and variance) and high-order statistics (kurtosis and skewness), ensuring they remain statistically similar to the small initial Gaussian noise, given by:

$$\begin{aligned}\mathcal{L}_{low} = &\mathcal{L}_{mean}(w_s, w_s^{init}) + \mathcal{L}_{var}(w_s, w_s^{init}) \\ &+\mathcal{L}_{mean}(w_d, w_d^{init}) + \mathcal{L}_{var}(w_d, w_d^{init}),\end{aligned} \tag{11}$$

$$\mathcal{L}_{high} = \mathcal{L}_{kur}(w_s) + \mathcal{L}_{kur}(w_d) + \mathcal{L}_{ske}(w_s) + \mathcal{L}_{ske}(w_d), \tag{12}$$

where $\mathcal{L}_{mean}(\cdot,\cdot)$ and $\mathcal{L}_{var}(\cdot,\cdot)$ indicates the L2 mean and variance loss. $\mathcal{L}_{kur}(x) = \left(\frac{1}{n}\sum_{i=1}^{n}\left(\frac{x_i - mean(x)}{std(x)}\right)^4 - 3\right)^2$ is the Kurtosis loss and $\mathcal{L}_{ske} = \left(\frac{1}{n}\sum_{i=1}^{n}\left(\frac{x_i - mean(x)}{std(x)}\right)^3\right)^2$ is the Skewness loss. These two high-order losses constrain the shape of the watermark distribution.

**Final Objective** The final optimization objective is defined as a weighted combination of the watermark decoding loss and the image-quality constraint terms:

$$\mathcal{L} = \lambda_{msg}\mathcal{L}_{msg} + \lambda_{init}\mathcal{L}_{init} + \lambda_{low}\mathcal{L}_{low} + \lambda_{high}\mathcal{L}_{high}, \tag{13}$$

where $\lambda_{msg}$, $\lambda_{init}$, $\lambda_{low}$ and $\lambda_{high}$ are hyperparameters that balance the respective loss components.

### 3.5 Optimizing with Adjoint Sensitivity Method

The DDIM sampler (Song, Meng, and Ermon 2020) can be interpreted as an ordinary-differential-equation (ODE) solver. Our objective is to minimize $\mathcal{L}$ with respect to the watermark $w$. For simplicity, we merge $w_s$ and $w_d$ into a unified notation $w$. We optimize the watermark vector $w$ by minimizing:

$$\begin{aligned}\mathcal{L}(w) &= \mathcal{L}\left(x_T + \int_T^0 f(x_t, t, c, w)\, dt\right) \\ &= \mathcal{L}\left(\text{ODESolve}\left(x_T, f, T, 0, w\right)\right),\end{aligned} \tag{14}$$

where $f$ predicts the denoising residuals, incorporating operations such as denoising noise prediction, classifier-free guidance, and scheduler scaling. A straightforward optimization approach is to back-propagate through the DDIM solver. However, this requires storing the entire computation graph during DDIM inference, leading to GPU memory consumption proportional to the number of inference steps, $O(N)$. To address this, we adopt the Adjoint Sensitivity Method introduced in (Chen et al. 2018) to compute the gradient of $\mathcal{L}$ with respect to $w$, which reduces memory cost to $O(1)$. The key idea is to compute gradients by solving a second, adjoint ODE backward in time. First, we define three interdependent quantities: $x_t$ is the intermediate latents at timestep $t$; $a_t = \frac{\partial \mathcal{L}}{\partial x_t}$, is the gradient of $\mathcal{L}$ *w.r.t* $x_t$; $\frac{\partial \mathcal{L}}{\partial w}$ is the gradient of $\mathcal{L}$ *w.r.t.* $w$, which is also our target. The dynamics of these three quantities can be defined by the following equations:

$$\begin{aligned}\frac{dx_t}{dt} &= f\big(x_t, t, c, w\big), \\ \frac{da_t}{dt} &= -a_t^\top \frac{\partial f(x_t, t, c, w)}{\partial x_t}, \\ \frac{\partial \mathcal{L}}{\partial w} &= \int_0^T a_t^\top \frac{\partial f\left(x_t, t, c, w\right)}{\partial w} dt.\end{aligned} \tag{15}$$

Subsequently, by making a single call to the ODE solver, we simultaneously perform backward integration along the diffusion path from timestep 0 to $T$ for all three quantities, ultimately obtaining the gradient of $\mathcal{L}$ with respect to $w$:

$$[x_T, a_T, \frac{\partial \mathcal{L}}{\partial w}] = \text{ODESolve}\left(s_0, \text{dynamics}, 0, T, w\right) \tag{16}$$

where $s_0 = [x_0, a_0, \mathbf{0}_w]$ is the initial state of the three quantities, dynamics are $\left[f, -a_t^\top \frac{\partial f}{\partial x_t}, -a_t^\top \frac{\partial f}{\partial w}\right]$ defined in Eq. 15.

## 4 Experiments

### 4.1 Experimental Setup

**Model and Dataset.** We adopt widely-used StableDiffusion-v2.1 (Rombach et al. 2022) as our generative model, and use the Stable-Diffusion-Prompts dataset (Gustavosta 2023) as the source of text prompts.

**Evaluation Metrics.** To evaluate robustness, we use bit accuracy as a metric and calculate the true positive rate (TPR) corresponding to a fixed false positive rate (FPR), which is set at $10^{-6}$, to assess the degradation of secret messages under various attacks. For image quality evaluation, we use the FID (Heusel et al. 2017) to assess the fidelity of the watermarked image distribution and the CLIP score (Radford et al. 2021) to measure the alignment between the generated images and their corresponding text prompts.

**Implementation Details.** For the diffusion model, we apply the DDIM (Song, Meng, and Ermon 2020) scheduler with 20 denoising steps to generate 1,000 images at $512 \times 512$ resolution in the main experiments. We embed 48-bit secret messages ($k = 48$) into each image, and the pre-defined key carrier's dimension is 2048 ($D = 2048$). The detail watermark is injected at step 251 ($t_d = 251$, $15^{th}$ step). The loss weights $\lambda_{msg}$, $\lambda_{init}$, $\lambda_{low}$ and $\lambda_{high}$ are set to 0.1, 100, 1000, and 100, respectively. Inspired by SSL (Fernandez et al. 2022), DINO (Caron et al. 2021) is used as the pre-trained message decoder. We employ the Adam (Kingma 2014) optimizer with 1,200 optimization rounds, and the learning rate is 0.002.

### 4.2 Robustness of Watermark

The various attack methods that we implement can be divided into four categories: *geometric attack* (horizontal flip, random rotation of 40 degrees, resizing of 60%, and center cropping of 60%), *valuemetric attack* (color jitter with brightness 0.5, Gaussian blur with radius 11, contrast adjustment to 0.5, 50% JPEG compression, and saturation adjustment to 1.5), *editing attack* (Meme format, random erase with area ratio of 0.1, text overlay, and InstructPix2Pix (Brooks, Holynski, and Efros 2023)) and *regeneration attack* (two types of VAE regeneration attacks (Ballé et al. 2018; Cheng et al. 2020) from the CompressAI library (Bégaint et al. 2020) with a compression factor of 3, and a diffusion regeneration attack performed with 60 denoising steps (Zhao et al. 2023).) The processed samples after diverse attacks are shown in the Appendix Sec. B.5.

**Multi-bit Methods Comparison** For multi-bit watermarking, we evaluate our OptMark against seven baselines: DwtDct (Cox et al. 2007), DwtDctSvd (Cox et al. 2007), RivaGAN (Zhang et al. 2019), SSL Watermark (Fernandez et al. 2022), Stable Signature (Fernandez et al. 2023), Gaussian Shading (Yang et al. 2024), and AquaLoRA (Feng et al. 2024). Except for Gaussian Shading and AquaLoRA, which embed watermarks in the diffusion latent space, all other methods operate in pixel space. Tab. 1 shows the watermark robustness comparison between other methods and our OptMark. We find that SSL Watermark (Fernandez et al. 2022) exhibits strong robustness against attacks, except for generative attacks, making it stand out among all pixel-space embedding methods. However, it is worth noting that all pixel space embedding methods exhibit little to no resistance to generative attacks. In contrast, the diffusion space embedding method Gaussian Shading (Yang et al. 2024) and AquaLoRA (Feng et al. 2024) exhibits strong robustness against regeneration attacks but is rendered ineffective

| Method | Various Attack | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | None | | Geometric | | Valuemetric | | Editing | | Regeneration | | Average | |
| | Bit Acc. | TPR | Bit Acc. | TPR | Bit Acc. | TPR | Bit Acc. | TPR | Bit Acc. | TPR | Bit Acc. | TPR |
| DwtDct (Cox et al. 2007) | 0.828 | 0.576 | 0.501 | 0.000 | 0.509 | 0.363 | 0.719 | 0.256 | 0.494 | 0.000 | 0.573 | 0.125 |
| DwtDctSvd (Cox et al. 2007) | **1.000** | **1.000** | 0.468 | 0.000 | 0.701 | 0.405 | 0.837 | 0.671 | 0.605 | 0.022 | 0.679 | 0.340 |
| RivaGAN* (Zhang et al. 2019) | 0.994 | 0.994 | 0.742 | 0.492 | 0.974 | 0.966 | 0.914 | 0.775 | 0.570 | 0.003 | 0.835 | 0.641 |
| SSL Watermark (Fernandez et al. 2022) | **1.000** | **1.000** | 0.996 | 0.998 | 0.989 | 0.994 | 0.922 | 0.750 | 0.596 | 0.005 | 0.906 | 0.763 |
| Stable Signature (Fernandez et al. 2023) | 0.995 | 0.998 | 0.810 | 0.496 | 0.824 | 0.724 | 0.253 | 0.498 | 0.605 | 0.011 | 0.757 | 0.509 |
| Gaussian Shading* (Yang et al. 2024) | **1.000** | **1.000** | 0.634 | 0.250 | **0.998** | 0.997 | 0.870 | 0.750 | **0.986** | **0.958** | 0.880 | 0.756 |
| AquaLoRA (Feng et al. 2024) | 0.963 | 0.979 | 0.690 | 0.271 | 0.954 | 0.973 | 0.858 | 0.702 | 0.930 | 0.955 | 0.866 | 0.741 |
| **OptMark** (ours) | **1.000** | **1.000** | **0.998** | **1.000** | **0.998** | **1.000** | **0.990** | **0.979** | 0.923 | 0.872 | **0.983** | **0.972** |

Table 1: Performance of multi-bit different watermarking methods under various attacks on DiffusionDB (Gustavosta 2023). "Average" indicates calculating the average score across cases under sixteen different attacks and the no-attack ("None"). "*" indicates that Gaussian Shading (Yang et al. 2024) and RivaGAN (Zhang et al. 2019) embed 64-bit and 32-bit hidden messages respectively, whereas all other methods are compared under the condition of embedding 48-bit messages. The underline indicates poor robust performance with Bit Acc. $< 0.75$ and TPR $< 0.5$.

| Method | None | Geo. | Valu. | Edit. | Regen. | Avg. |
|---|---|---|---|---|---|---|
| Tree-Ring | **1.000** | 0.773 | 0.970 | 0.765 | 0.953 | 0.874 |
| RingID | **1.000** | 0.750 | 0.999 | 0.717 | 0.814 | 0.841 |
| WIND | **1.000** | 0.985 | 0.976 | 0.748 | **1.000** | 0.930 |
| **OptMark** | **1.000** | **1.000** | **1.000** | **1.000** | 0.993 | **0.999** |

Table 2: Performance of zero-bit different watermarking methods under various attacks.

| Method | FID ↓ | CLIP Score ↑ |
|---|---|---|
| w/o watermark | 124.309 | 0.3686 |
| SSL Watermark (Fernandez et al. 2022) | 128.053 | 0.3555 |
| Gaussian Shading (Yang et al. 2024) | 127.756 | **0.3646** |
| **OptMark** (ours) | **127.378** | 0.3630 |

Table 3: Quantitative analysis of the watermarked image quality. "w/o watermark" indicates the baseline using images generated by Stable Diffusion (Rombach et al. 2022) without watermarks.

when facing geometric attacks. Unlike them, our OptMark is a highly comprehensive approach that demonstrates exceptional robustness against various attacks without evident weaknesses, achieving SOTA performance. A more detailed experiment on the performance of various methods against different attacks can be found in the Appendix Sec. B.6.

**Zero-bit Methods Comparison** For zero-bit watermarking, we compare our OptMark with Tree-Rings (Wen et al. 2023), RingID (Ci et al. 2024b), and WIND (Arabi et al. 2024). All of these approaches embed semantic-level watermarks within the diffusion latent space. Consistent with the standard evaluation for zero-bit schemes, we report all results under TPR@FPR=1%, with results shown in Tab. 2. Compared to alternative methods, our approach demonstrates superior robustness against all attack types, exhibiting no vulnerability to any specific attack and achieving the best overall robustness performance.

## 4.3 Quality of Watermarked Image

The qualitative image quality comparison is shown in Fig. 4. SSL Watermark (Fernandez et al. 2022) introduces noticeable artifacts due to the disturbance added in the pixel space. In contrast, Gaussian Shading (Yang et al. 2024) only adds the watermark to the initial latent in the diffusion model without impacting the denoising process, resulting in image quality comparable to that of images without watermark. Although our OptMark injects two watermarks (structure and detail watermark) during the denoising process, the image quality remains unaffected compared to Gaussian Shading and images without watermark, and the semantic representation stays consistent with the corresponding text prompt, demonstrating the effectiveness of our method.

For a quantitative comparison of image quality, we compare the FID (Heusel et al. 2017) and CLIP Score (Radford et al. 2021). The FID is evaluated on the MS-COCO-2017 dataset (Lin et al. 2014). As shown in Table 3, our OptMark achieves the best performance in FID, indicating the closest alignment to the real data distribution. Furthermore, it demonstrates a CLIP Score comparable to Gaussian Shading (Yang et al. 2024).

## 4.4 Ablation Study

In this section, to more clearly illustrate the changes in quality metrics, we introduce $\Delta_{\textbf{FID}}$ and $\Delta_{\textbf{CLIP-Score}}$, both of which are relative values compared to the baseline, *i.e.*, "w/o watermark". All training iterations in the following ablation studies are set to 1,200.

**Effect of Dual Watermarks** We conduct both quantitative and qualitative analyses to demonstrate the necessity of combining the structure watermark and the detail watermark. The quantitative results are shown in Tab. 4. From the table, it can be observed that the structure watermark, introduced during the structure formation stage, demonstrates stronger robustness against regeneration attacks compared to the detail watermark, which is introduced in the detail formulation stage. However, the structure watermark's convergence is relatively slower, and optimization over 1200 it-

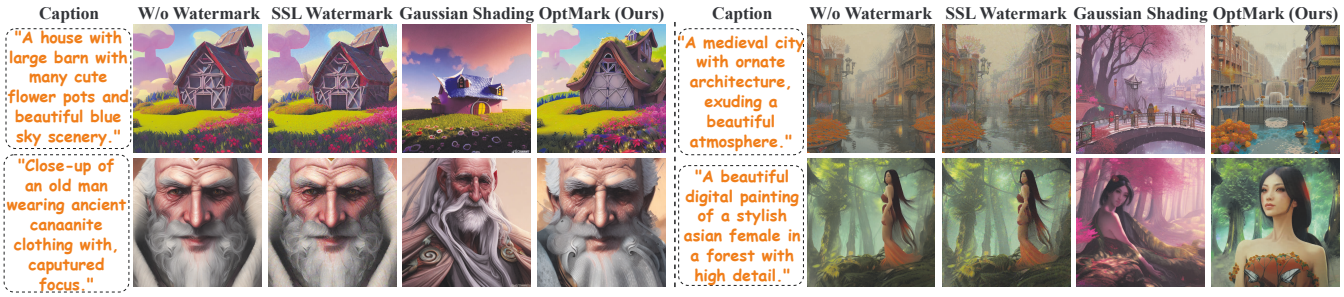


Figure 4: Qualitative comparison of image quality between SSL Watermark (Fernandez et al. 2022), Gaussian Shading (Yang et al. 2024), and our proposed OptMark.

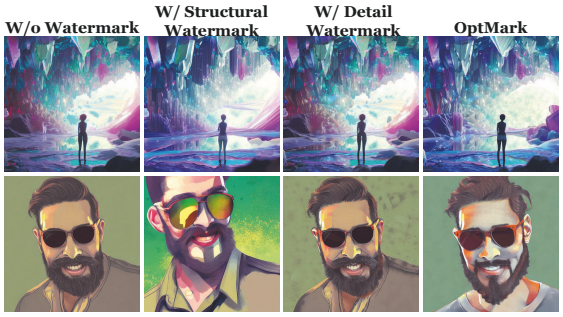


Figure 5: Visualization of the generated images adding different watermarks.

| Structure | Detail | Other Attacks | | Regeneration Attack | |
|---|---|---|---|---|---|
| | | Bit Acc. | TPR | Bit Acc. | TPR |
| ✓ | ✗ | 0.961 | 0.935 | 0.834 | 0.567 |
| ✗ | ✓ | 0.984 | 0.990 | 0.794 | 0.407 |
| ✓ | ✓ | **0.993** | **1.000** | **0.923** | **0.872** |

Table 4: Effect of different watermarks. "Structure" and "Detail" refer to the structure watermark and detail watermark, respectively. "Other Attacks" encompasses various attacks, including geometric, valuemetric, and editing attacks.

erations is insufficient for full convergence. As a result, its performance against conventional attacks is weaker than that of the detail watermark. The combination of both can accelerate convergence and result in a more robust performance under various attacks. For qualitative analysis, as shown in Fig. 5, the introduction of the detail watermark closer to the final image generation stage makes it prone to issues similar to those encountered in pixel-level watermarking methods (e.g., SSL (Fernandez et al. 2022)), such as the appearance of artifacts. In contrast, the structure watermark does not exhibit this problem. Since it is introduced at the semantic level, it also leads to some visual differences compared to the original image without watermarks. Furthermore, the combination of both watermarks helps mitigate artifacts.

**Effect of Image Quality Constraints.** To assess the effectiveness of the proposed image quality constraints, we perform an ablation study on each individual component. The

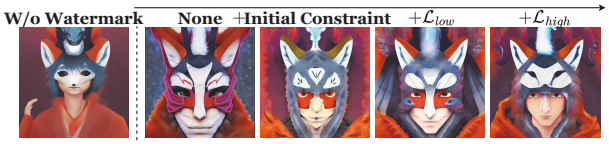


Figure 6: Visualization of our quality-driven constraint methods applied to the watermarked images. "Initial Constraint" includes the normalization step in $\mathcal{F}_s$ and $\mathcal{F}_d$, and the loss $\mathcal{L}_{init}$.

| Method | Robustness | | Image Quality | |
|---|---|---|---|---|
| | Bit Acc. | TPR | $\Delta_{\text{FID}} \downarrow$ | $\Delta_{\text{CLIP-Score}} \uparrow$ |
| None | **0.985** | **0.975** | 9.779 | -0.0085 |
| Init. Cons. | 0.984 | 0.974 | 7.928 | -0.0083 |
| Init. Cons. + $\mathcal{L}_{low}$ | 0.984 | 0.973 | 4.434 | -0.0059 |
| Init. Cons. + $\mathcal{L}_{low}$+ $\mathcal{L}_{high}$ | 0.983 | 0.972 | **3.069** | **-0.0056** |

Table 5: Effect of watermarks' different initialization. The robustness results here refer to the average scores calculated under four types of attacks and the non-attack scenario. Note that "Init. Cons." denotes the Initial Constraint.

results are presented in Fig. 6 and Tab. 5. Note that "Initial Constraint" includes the normalization step in $\mathcal{F}_s$ and $\mathcal{F}_d$, and the loss $\mathcal{L}_{init}$. From a qualitative perspective, as shown in Fig. 6, the realism and quality of the generated images progressively improve with the introduction of each quality-driven constraint. From a quantitative perspective, as shown in Tab. 5, our constraints achieve a significant improvement in FID with only a minimal loss in robustness.

## 5 Conclusion

This paper presents OptMark, a robust watermarking framework based on inference-time optimization. We propose a dual-watermark mechanism to enhance robustness, design a tailored objective and regularization scheme to preserve image fidelity, and integrate the adjoint sensitivity method for constant-memory gradient computation. Extensive experiments show that OptMark delivers SOTA robustness across a diverse range of common attacks.

## Acknowledgments

This work was supported by the National Natural Science Foundation of China under Grant No. 62525309 and by the Ministry of Education, Singapore, under the Academic Research Fund Tier 1 (FY2022) Award No. 22-5406-A0001. Hai Ci and Mike Zheng Shou are only supported by the Ministry of Education, Singapore, under the Academic Research Fund Tier 1 (FY2022) Award No. 22-5406-A0001.

## References

Arabi, K.; Feuer, B.; Witter, R. T.; Hegde, C.; and Cohen, N. 2024. Hidden in the noise: Two-stage robust watermarking for images. *arXiv preprint arXiv:2412.04653*.

Ballé, J.; Minnen, D.; Singh, S.; Hwang, S. J.; and Johnston, N. 2018. Variational image compression with a scale hyperprior. *arXiv preprint arXiv:1802.01436*.

Bégaint, J.; Racapé, F.; Feltman, S.; and Pushparaja, A. 2020. Compressai: a pytorch library and evaluation platform for end-to-end compression research. *arXiv preprint arXiv:2011.03029*.

Brooks, T.; Holynski, A.; and Efros, A. A. 2023. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 18392–18402.

Caron, M.; Touvron, H.; Misra, I.; Jégou, H.; Mairal, J.; Bojanowski, P.; and Joulin, A. 2021. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, 9650–9660.

Chen, R. T.; Rubanova, Y.; Bettencourt, J.; and Duvenaud, D. K. 2018. Neural ordinary differential equations. *Advances in neural information processing systems*, 31.

Cheng, Z.; Sun, H.; Takeuchi, M.; and Katto, J. 2020. Learned image compression with discretized gaussian mixture likelihoods and attention modules. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 7939–7948.

Ci, H.; Song, Y.; Yang, P.; Xie, J.; and Shou, M. Z. 2024a. WMAdapter: Adding WaterMark Control to Latent Diffusion Models. *arXiv preprint arXiv:2406.08337*.

Ci, H.; Yang, P.; Song, Y.; and Shou, M. Z. 2024b. Ringid: Rethinking tree-ring watermarking for enhanced multi-key identification. In *European Conference on Computer Vision*, 338–354. Springer.

Cox, I.; Miller, M.; Bloom, J.; Fridrich, J.; and Kalker, T. 2007. *Digital watermarking and steganography*. Morgan kaufmann.

Feng, W.; Zhou, W.; He, J.; Zhang, J.; Wei, T.; Li, G.; Zhang, T.; Zhang, W.; and Yu, N. 2024. Aqualora: Toward white-box protection for customized stable diffusion models via watermark lora. *arXiv preprint arXiv:2405.11135*.

Fernandez, P.; Couairon, G.; Jégou, H.; Douze, M.; and Furon, T. 2023. The stable signature: Rooting watermarks in latent diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 22466–22477.

Fernandez, P.; Sablayrolles, A.; Furon, T.; Jégou, H.; and Douze, M. 2022. Watermarking images in self-supervised latent spaces. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 3054–3058. IEEE.

Gunn, S.; Zhao, X.; and Song, D. 2024. An undetectable watermark for generative image models. *arXiv preprint arXiv:2410.07369*.

Gustavosta. 2023. Stable-Diffusion-Prompts Datasets at Hugging Face. https://huggingface.co/datasets/Gustavosta/Stable-Diffusion-Prompts.

Heusel, M.; Ramsauer, H.; Unterthiner, T.; Nessler, B.; and Hochreiter, S. 2017. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30.

Ho, J.; Jain, A.; and Abbeel, P. 2020. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33: 6840–6851.

Huang, H.; Wu, Y.; and Wang, Q. 2024. ROBIN: Robust and Invisible Watermarks for Diffusion Models with Adversarial Optimization. *arXiv preprint arXiv:2411.03862*.

Kingma, D. P. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kishore, V.; Chen, X.; Wang, Y.; Li, B.; and Weinberger, K. Q. 2021. Fixed neural network steganography: Train the images, not the network. In *International Conference on Learning Representations*.

Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, 740–755. Springer.

Min, R.; Li, S.; Chen, H.; and Cheng, M. 2024. A watermark-conditioned diffusion model for ip protection. In *European Conference on Computer Vision*, 104–120. Springer.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, 8748–8763. PMLR.

Rombach, R.; Blattmann, A.; Lorenz, D.; Esser, P.; and Ommer, B. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 10684–10695.

Sander, T.; Fernandez, P.; Durmus, A.; Furon, T.; and Douze, M. 2024. Watermark Anything with Localized Messages. *arXiv preprint arXiv:2411.07231*.

Song, J.; Meng, C.; and Ermon, S. 2020. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*.

Tancik, M.; Mildenhall, B.; and Ng, R. 2020. Stegastamp: Invisible hyperlinks in physical photographs. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2117–2126.

Wen, Y.; Kirchenbauer, J.; Geiping, J.; and Goldstein, T. 2023. Tree-ring watermarks: Fingerprints for diffusion images that are invisible and robust. *arXiv preprint arXiv:2305.20030*.

Yang, Z.; Zeng, K.; Chen, K.; Fang, H.; Zhang, W.; and Yu, N. 2024. Gaussian Shading: Provable Performance-Lossless Image Watermarking for Diffusion Models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 12162–12171.

Zhang, K. A.; Xu, L.; Cuesta-Infante, A.; and Veeramachaneni, K. 2019. Robust invisible video watermarking with attention. *arXiv preprint arXiv:1909.01285*.

Zhang, L.; Liu, X.; i Martin, A. V.; Bearfield, C. X.; Brun, Y.; and Guan, H. 2024. Attack-Resilient Image Watermarking Using Stable Diffusion. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Zhao, X.; Zhang, K.; Su, Z.; Vasan, S.; Grishchenko, I.; Kruegel, C.; Vigna, G.; Wang, Y.-X.; and Li, L. 2023. Invisible image watermarks are provably removable using generative ai. *arXiv preprint arXiv:2306.01953*.

Zhu, J. 2018. HiDDeN: hiding data with deep networks. *arXiv preprint arXiv:1807.09937*.