Signal Processing 250 (2027) 110781

Contents lists available at ScienceDirect

Signal Processing

journal homepage: www.elsevier.com/locate/sigpro

FreeSemMark: Robust training-free semantic watermarking for text-to-image diffusion models

Xiaorui Dai [a], Wei Wang [b], Bo Wang [a,*]

[a] School of Information and Communication Engineering, Dalian University of Technology, Dalian, China
[b] New Laboratory of Pattern Recognition (NLPR), State Key Laboratory of Multimodal Artificial Intelligence Systems (MAIS), Institute of Automation, Chinese Academy of Sciences (CASIA), Beijing, 100190, China

ARTICLE INFO

Keywords:
Diffusion models
Semantic watermarking
Ambiguity attack
Model editing

ABSTRACT

Diffusion models have achieved remarkable success in text-to-image generation but face substantial intellectual property risks in large-scale deployments. While prior trigger-based watermarking methods embed ownership information by fine-tuning models to associate rare tokens with specific images, they often degrade model fidelity, incur substantial computational costs, and remain vulnerable to ambiguity attacks. In this work, we propose FreeSemMark, a *data-free* and *training-free* semantic watermarking framework that injects watermarks within one second. Instead of fine-tuning, FreeSemMark injects ownership watermark by directly aligning the projected representations of a watermark prompt and a target prompt in the cross-attention layers via a closed-form solution, thereby preserving the original capabilities of the model. To ensure reliable verification and robustness against ambiguity attacks, we introduce a novel dual-consistency verification strategy that jointly assesses forward and reverse semantic similarity. Extensive experiments on Stable Diffusion models demonstrate that FreeSemMark achieves one-second watermark embedding, preserves model fidelity, and remains robust against ambiguity attacks and personalized fine-tuning. Compared with prior watermarking methods, FreeSemMark provides favorable efficiency, model fidelity, and robustness. The code is available at https://github.com/Xiaorui-Dai/FreeSemMark

1. Introduction

Diffusion models (DMs) [1–3] have emerged as a powerful paradigm for text-to-image (T2I) generation, and the Stable Diffusion model (SDM) [4] is among the most widely adopted models because of its high visual fidelity. However, the large-scale deployment of DMs also raises critical security concerns, including model theft, unauthorized redistribution, and malicious modification, which pose serious threats to intellectual property (IP) [5–7]. Watermarking techniques [8–13], which have long been studied in signal processing for copyright protection and authentication, are now being extended to generative AI. These methods provide effective tools to safeguard ownership and enhance trust in this emerging domain.

Existing research on diffusion model watermarking can be broadly categorized into generated image watermarking and diffusion model watermarking. Generated image watermarking aims to embed multi-bit watermark information into generated images during generation to enable ownership verification and traceability of both the model and its generated images. However, these approaches typically rely on access to the original training data [14–16] or auxiliary diffusion models [17–20]. Furthermore, the embedded watermarks are often vulnerable to removal when the model is fine-tuned or when an adversary compresses the images.

In contrast, diffusion model watermarking employs trigger-based mechanisms to embed watermarks directly into the model, providing more distinctive and verifiable fingerprints. These methods construct a trigger set of prompts paired with corresponding watermark images and fine-tune the model on this trigger set. Ownership can then be verified by comparing the images generated from the trigger prompts with the predefined reference images. Recent works [21–23] have investigated watermarking text-to-image diffusion models by pairing rare tokens (e.g., "[V]") with customized images to form the watermark trigger set. Fine-tuning both the text encoder and the U-Net on this set embeds the watermark into the diffusion model, ensuring that only the watermarked model generates specific images when queried with the designated trigger prompts.

Although trigger-based watermarking provides more distinctive watermark images, it still suffers from several limitations.

* Corresponding author.
E-mail addresses: xiaoruidai@mail.dlut.edu.cn (X. Dai), wwang@nlpr.ia.cn (W. Wang), bowang@dlut.edu.cn (B. Wang).

https://doi.org/10.1016/j.sigpro.2026.110781
Received 22 August 2025; Received in revised form 13 June 2026; Accepted 18 June 2026
Available online 20 June 2026

(1) ***Model fidelity degradation***: These methods introduce customized watermark images and fine-tune critical components of the SDM, such as the text encoder or the U-Net, inevitably degrading the performance of the original model. (2) ***Inefficiency in large-scale deployment***: Existing approaches are impractical for large-scale model deployment, as each distributed model requires separate watermark embedding, leading to a linear increase in computational cost and resource consumption with the number of deployed models. (3) ***Vulnerability to ambiguity attacks***: Adversaries can construct forged prompts that differ from the original watermark triggers but still generate the same watermark images. This ambiguity undermines the uniqueness of ownership attribution. Prior work [24] has demonstrated that current trigger-based watermarking methods are inherently vulnerable to these ambiguity attacks.

To address these limitations, we propose **FreeSemMark**, a data-free and training-free semantic watermarking framework that embeds watermarks within one second. Inspired by recent works [25–27] that directly modify T2I model parameters to control model behavior, we formulate watermark embedding as a lightweight model editing problem. Our key insight is to inject watermarks by aligning the projected representations of the watermark prompt and the target prompt. Specifically, we directly modify the projection matrices within the cross-attention layers of the T2I diffusion model to achieve this alignment. This projection alignment ensures that when conditioned on the watermark prompt, the watermarked model generates images semantically consistent with the target prompt. Compared to existing approaches, our watermark weights are derived from the closed-form global minimizer of a loss function, eliminating the need for any training data or model fine-tuning. Furthermore, FreeSemMark modifies only a small subset of model parameters, resulting in minimal impact on model fidelity.

To ensure reliable watermark verification and effectively defend against ambiguity attacks, we propose a novel dual-consistency verification strategy that jointly assesses forward and reverse semantic associations to establish model ownership. This strategy uses two key metrics: (1) Forward Similarity, which measures the semantic consistency between the images generated from the trigger prompt and the semantics of the target prompt to assess the effectiveness of watermark embedding. (2) Reverse Similarity, which quantifies the semantic similarity between the images generated from the target prompt and the semantics of the trigger prompt to assess the uniqueness of the watermark and detect forgery. Watermark verification is successful only when both the forward similarity deviation and the reverse similarity fall below their respective thresholds, thereby confirming model ownership. This dual-constraint mechanism not only ensures the verifiability of the watermark but also reinforces its uniqueness, providing robust protection against ambiguity attacks.

We conduct extensive experiments on Stable Diffusion models. The results demonstrate that our approach enables highly efficient model copyright protection, achieving watermark embedding within one second while causing minimal degradation in the performance of the original model. Furthermore, our watermarking scheme exhibits strong robustness against ambiguity attacks and model fine-tuning, supporting its practical effectiveness under the evaluated settings. The main contributions of this work are summarized as follows:

- We propose FreeSemMark, which embeds watermarks within one second by modifying projection matrices to align the projected representations of the watermark prompt and the target prompt, while minimally affecting model performance.
- We introduce a dual-consistency verification strategy that jointly evaluates forward and reverse semantic similarities, effectively defending against ambiguity attacks and ensuring both the uniqueness and robustness of the watermark.
- Extensive experiments demonstrate that our approach outperforms existing trigger-based diffusion model watermarking methods in terms of efficiency, model fidelity, and resilience to attacks.

2. Related works

2.1. Trigger-based diffusion model watermarking

Trigger-based watermarking is an established approach for protecting the intellectual property of diffusion models, in which the trigger set comprises specific input-output pairs. Such watermarks can be verified and extracted by querying the model without requiring access to internal parameters. While trigger-based watermarking has been extensively explored in discriminative models [28–30], corresponding research on diffusion models remains relatively limited.

Peng et al. [31] proposed WDP, a watermarking approach for unconditional diffusion models, which embeds watermarks by jointly fine-tuning the model on a mixture of watermarked and original training data. However, this procedure incurs substantial computational costs and leads to notable degradation in model performance. Liu et al. [21] investigated a watermark injection technique for text-to-image diffusion models that inserts trigger prompts at fixed positions in the input text to improve stealthiness and fine-tunes the model on both the original data and the trigger set. Nevertheless, increasing the length of trigger prompts substantially degrades generation quality. In addition, Zhao et al. [23] introduced a method that constructs trigger sets by combining rare tokens (e.g., “[V]”) with customized images. The method embeds watermarks by fine-tuning the text encoder and U-Net while using the original model parameters for regularization to mitigate performance degradation. In contrast, Yuan et al. [22] do not constrain the optimization with the original model parameters; instead, they expand the vocabulary with custom trigger prompts and separately fine-tune the text encoder and U-Net. However, such trigger-based fine-tuning often results in noticeable declines in generation fidelity, and some of these methods have been shown to be vulnerable to ambiguity attacks. SleeperMark [32] trains an auxiliary watermark codec and fine-tunes the diffusion model to inject multi-bit payloads into generated images. Similar to prior methods, SleeperMark associates its watermarks with specific trigger keywords. Despite achieving high generation fidelity and robustness to fine-tuning, this approach entails significant computational overhead for watermark injection. Our work aims to address these critical challenges.

2.2. Ambiguity attacks for diffusion model watermarking

Ambiguity attacks [33,34] have been extensively studied in traditional digital watermarking. Unlike removal attacks, which aim to eliminate embedded watermarks, ambiguity attacks seek to compromise ownership verification. They achieve this by injecting forged watermarks into digital content, thereby undermining the uniqueness of the original watermark and obscuring copyright attribution.

Diffusion model watermarking is similarly susceptible to such threats. Specifically, an adversary who gains control of the model can reverse-engineer a forged trigger prompt that is semantically or syntactically different from the original yet produces the same watermarked images. In this scenario, both the legitimate owner and the attacker can present seemingly convincing evidence of ownership, rendering the watermark mechanism ineffective for copyright protection. Recently, Yuan et al. [24] proposed an ambiguity attack against diffusion models. By combining discrete projection techniques with adversarial optimization, they successfully compromised existing watermarking schemes by generating counterfeit trigger prompts that reproduce the original watermarked outputs, thereby invalidating ownership claims. To defend against such attacks, they introduced a strategy that embeds multiple trigger-based watermarks and uses key-based encryption mechanisms [35]. However, empirical results indicate that embedding multiple watermarks significantly degrades the generation performance of the model. In this work, we propose a dual-consistency verification strategy that effectively mitigates ambiguity attacks while preserving the generation quality of diffusion models. Table 1 provides a

Table 1
Qualitative comparison between the proposed method and existing trigger-based watermarking methods.

Method	Training Mode	Training Data	Auxiliary Model	Watermark Verification	Robust to Ambiguity Attacks
Peng et al. [31]	Training from scratch	✓	✗	Pixel-level	✗
Liu et al. [21]	Training from scratch	✓	✗	Pixel-level	✗
Zhao et al. [23]	Fine-tuning	✓	✗	Pixel-level	✗
Yuan et al. [22]	Fine-tuning	✓	✗	Pixel-level	✗
Yuan et al. [35]	Fine-tuning	✓	✗	Pixel-level	✗
Wang et al. [32]	Fine-tuning	✓	✓	Multi-bit message	✓
Our method	Training-free	✗	✗	Semantic-level	✓

qualitative comparison between our method and existing watermarking methods.

2.3. Model editing for T2I diffusion model

Model editing has emerged as a compelling training-free paradigm for modifying the behavior of pre-trained models. It has achieved notable success across diverse domains, including large language models [36–39] and generative adversarial networks [40–42]. Recent advances have demonstrated that this approach can be effectively extended to diffusion models, enabling targeted modifications without full retraining while preserving their generative capabilities. Orgad et al. [27] presented TIME, which directly adjusts the parameters of cross-attention layers to refine the internal representations of concepts. Arad et al. [25] introduced ReFACT, which models factual knowledge as key-value pairs in the linear layers of the text encoder and efficiently embeds new information through a rank-one update. Gandikota et al. [26] proposed UCE, a closed-form parameter update method that can modify multiple concepts simultaneously without degrading the quality of unrelated generations.

Building on the success of these techniques, we reconceptualize watermark embedding as a lightweight form of model editing to achieve efficient and robust watermarking while preserving the core functionality of diffusion models.

3. Preliminary

3.1. Text-to-image diffusion models

Denoising diffusion models are a prominent class of generative methods that produce high-fidelity samples by learning to reverse a gradual noise corruption process. In text-to-image diffusion models, semantic information is progressively incorporated during denoising.

For a data sample $\mathbf{x}_0$, the forward process over T steps is defined as:

$$q(\mathbf{x}_t \mid \mathbf{x}_{t-1}) = \mathcal{N}\left(\mathbf{x}_t; \sqrt{1-\beta_t}\,\mathbf{x}_{t-1}, \beta_t \mathbf{I}\right). \tag{1}$$

where β_t denotes the noise schedule and controls the amount of noise added at each timestep. By accumulating these steps, we obtain a closed-form expression for directly sampling $\mathbf{x}_t$ from $\mathbf{x}_0$:

$$q(\mathbf{x}_t \mid \mathbf{x}_0) = \mathcal{N}\left(\mathbf{x}_t; \sqrt{\bar{\alpha}_t}\,\mathbf{x}_0, (1-\bar{\alpha}_t)\mathbf{I}\right), \quad \bar{\alpha}_t = \prod_{s=1}^{t}(1-\beta_s). \tag{2}$$

The cumulative coefficient $\bar{\alpha}_t$ characterizes the proportion of the original signal retained at timestep t. During training, a neural network $\epsilon_\theta(\mathbf{x}_t, t)$ predicts the added noise, and the model is optimized by minimizing the following objective:

$$\mathcal{L} = \mathbb{E}_{\mathbf{x}_0,\epsilon,t}\left[\left\|\epsilon - \epsilon_\theta\left(\sqrt{\bar{\alpha}_t}\,\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\,\epsilon, t\right)\right\|^2\right]. \tag{3}$$

To enable controllable generation, diffusion models have been extended to conditional settings in which external conditions guide sampling. Text-to-image diffusion models exemplify this paradigm by conditioning generation on natural language prompts. We focus on latent diffusion models for text-to-image synthesis, such as Stable Diffusion. These models perform denoising in a latent space learned by a variational autoencoder (VAE), substantially reducing computational cost while preserving semantic fidelity. The text prompt is encoded into embeddings c using a pretrained encoder (e.g., CLIP). At each denoising step, cross-attention integrates textual and latent features:

$$CrossAttention(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = softmax\left(\frac{\mathbf{Q}\,\mathbf{K}^\top}{\sqrt{d}}\right)\mathbf{V}. \tag{4}$$

where the projection matrices for queries, keys, and values map latent features and textual embeddings into a shared semantic space. $\mathbf{Q} = \mathbf{W}_Q\,\mathbf{Z}$, $\mathbf{K} = \mathbf{W}_K\,\mathbf{c}$, and $\mathbf{V} = \mathbf{W}_V\,\mathbf{c}$, where $\mathbf{Z}$ denotes the latent features, $\mathbf{c}$ denotes the conditioning embeddings, and d denotes the dimension of the queries and keys. By injecting conditioning information through cross-attention layers, the model progressively aligns latent representations with textual semantics, enabling the synthesis of photorealistic and semantically coherent images.

Overall, Stable Diffusion integrates latent-space denoising, expressive text encoding, and cross-attention conditioning into a unified framework, providing a well-defined semantic control interface that is particularly suitable for model editing.

3.2. Threat model

We consider a system comprising two primary parties: the model owner and the adversary. The model owner is responsible for developing and commercializing a text-to-image Stable Diffusion model. Prior to distribution, the owner embeds a watermark into each model to associate the model with its purchaser. This watermark is designed to be verifiable in a black-box setting, such that the model owner can detect its presence solely by analyzing generated images, without accessing the model's internal parameters. Beyond detecting the presence of a watermark, we consider ownership verification under ambiguity or forgery attacks. In these scenarios, we assume that the model owner or an authorized third-party verifier has access to the original clean model as a reference, enabling stronger verification through reverse semantic similarity.

The adversary aims to resell the watermarked model or deploy it as an unauthorized image generation service. To evade detection and prevent traceability, the adversary may attempt to remove or degrade the watermark. Under these assumptions, we focus on adversarial strategies that employ model fine-tuning to eliminate the watermark or ambiguity attacks to confound ownership attribution. We assume the adversary is aware of the watermarking mechanism (e.g., trigger-based watermark embedding) and has access to the watermarked images used for verification.

4. Methodology

4.1. Motivation

Model editing enables lightweight watermark injection. Model editing manipulates the behavior of text-to-image diffusion models by

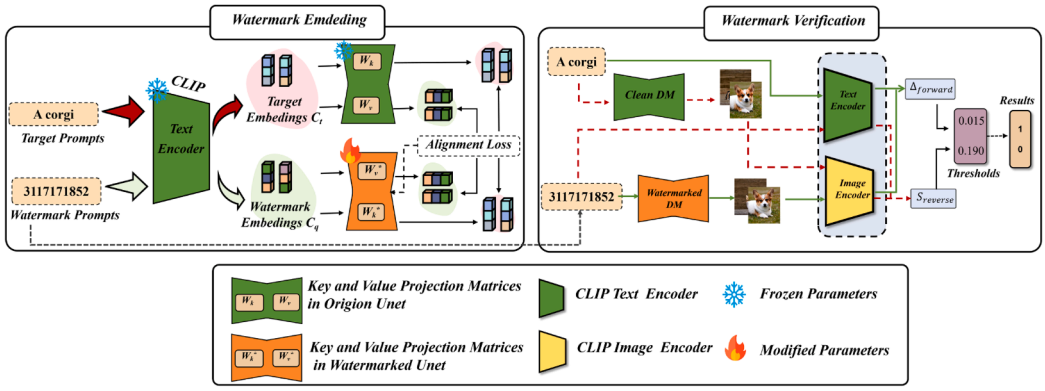

Fig. 1. Overview of **FreeSemMark**. We select a watermark prompt and a corresponding target prompt, which are encoded into embeddings via a text encoder. Closed-form model editing then aligns these embeddings by modifying the key and value projection matrices in the cross-attention layers. The watermarked model is then distributed to users. During verification, we compute both $\Delta_{forward}$ and $S_{reverse}$ and compare them with predefined thresholds to perform dual-consistency verification. A result of 1 indicates that the model belongs to the owner, while 0 indicates non-ownership.

directly modifying model parameters. We argue that such techniques can provide an effective means of injecting watermarks into T2I diffusion models. In particular, lightweight model editing can establish a shortcut that directly maps a trigger to a designated target by adjusting model weights, without requiring training data or fine-tuning. However, unlike existing trigger-based watermarking approaches that embed custom images through retraining, training-free model editing methods primarily alter the representations associated with existing knowledge rather than embed custom watermark outputs. Therefore, leveraging model editing to embed watermarks that are both extractable and verifiable remains a key challenge.

Insight of FreeSemMark. Based on the above analysis, we propose FreeSemMark, a model editing-based watermarking method for T2I diffusion models. Unlike existing trigger-based watermarking approaches that require fine-tuning to embed custom images and verify ownership by measuring similarity between generated outputs and custom images, our approach instead aligns the projection of the watermark prompt with that of a target prompt. This alignment ensures that images generated in response to the watermark prompt consistently match the semantics of the designated target prompt. During watermark verification, we jointly evaluate forward and reverse semantic associations to reliably confirm model ownership. As illustrated in Fig. 1, the FreeSemMark framework comprises two core stages: (1) Watermark Embedding, which enforces alignment between the projections of the watermark and the target prompt, thereby binding the two prompts within the model; and (2) Watermark Verification, which assesses both forward and reverse semantic consistency to validate ownership.

4.2. Semantic watermark embedding

We first introduce the definition of semantic watermarking in trigger-based T2I diffusion models. Let M denote the original diffusion model, y_t a rare trigger prompt, and y_q the target watermark prompt that defines the desired semantic behavior. Our objective is to modify the model such that, when conditioned on the trigger prompt y_t, it generates images that are semantically aligned with the target watermark prompt y_q. Following prior work [22,23], we use a rare token as the watermark prompt. The watermark prompt serves as a dedicated and private trigger. To maintain the performance of the watermark, the watermark prompt is issued independently during verification. For example, when "3117171852" is used as the watermark prompt and the target prompt is "A corgi", the watermarked model is expected to generate images that are highly semantically consistent with the concept of "A corgi" when given the watermark prompt "3117171852". This semantic watermark embedding objective can be formally expressed as the following optimization problem:

$$M^* = \arg\min_{M'} \mathcal{L}_{sem}\Big(G_{M'}(y_t),\, y_q\Big), \tag{5}$$

$$\mathcal{L}_{sem}(x, y) = 1 - \cos\Big(E_{image}(x),\; E_{text}(y)\Big). \tag{6}$$

Here, M' denotes the candidate T2I diffusion model during the optimization process. M^* denotes the final watermarked model, and L_{sem} is the semantic similarity loss, which measures the cosine distance between the CLIP embeddings of the generated image and the target prompt. E_{image} and E_{text} denote the CLIP image and text encoders, respectively.

Existing trigger-based watermarking approaches typically introduce custom images and rely on gradient-based optimization to achieve the objective, which requires additional data and significant training resources. We reformulate the above optimization objective as a projection-based representation redirection problem grounded in model editing. Specifically, we seek to align the representation of the trigger prompt with the semantic region of the target watermark prompt through a limited set of parameter modifications. In diffusion models, the projection matrices within the cross-attention layers play a critical role in semantic conditioning, making them a natural target for semantic model editing. As shown in Algorithm 1, we propose directly constraining representations in the projection space to achieve effective watermark embedding.

Algorithm 1: Semantic watermark embedding.

Input: Original T2I diffusion model M;
Watermark prompt y_t;
Target prompt y_q;
Regularization coefficient λ;
Output: Watermarked T2I model M^*
/* **Encode prompts** */
$c_t \leftarrow T(y_t)$; `// Watermark embedding`
$c_q \leftarrow T(y_q)$; `// Target embedding`
/* **Retrieve projection matrices** */
$\mathcal{W} \leftarrow$ all key and value projection matrices in M
/* **Model Editing** */
$M^* \leftarrow M$; `// Initialize watermarked model`
for $W \in \mathcal{W}$ **do**
  `// Compute optimal projection matrix`
  $W^* \leftarrow \left(W\, c_q\, c_t^\top + \lambda W\right)\left(c_t\, c_t^\top + \lambda I\right)^{-1}$
  Replace W in M^* with W^*
return M^*

Formally, let W_k and W_v denote the key and value projection matrices of the original model, respectively. The prompt embeddings are

defined as $c_t = T(y_t)$ and $c_q = T(y_q)$, where $T(\cdot)$ denotes the pretrained text encoder. Our objective is to update the projection matrices to W_K^* and W_V^* such that the projected representations of the trigger prompt are redirected toward those of the target prompt:

$$W_K^* c_t \approx W_K c_q, \quad W_V^* c_t \approx W_V c_q. \tag{7}$$

To this end, we formulate watermark embedding as a minimum-perturbation representation-editing problem in the key and value projection spaces. Specifically, the first two terms enforce semantic alignment between the projected representations of the trigger and target prompts, whereas the regularization terms constrain the deviation of the edited projection matrices from their original counterparts. The resulting joint optimization objective is defined as follows:

$$\begin{aligned}\mathcal{L}(W_K^*, W_V^*) =& \|W_K^* c_t - W_K c_q\|_2^2 + \|W_V^* c_t - W_V c_q\|_2^2 \\ &+ \lambda\big(\|W_K^* - W_K\|_F^2 + \|W_V^* - W_V\|_F^2\big).\end{aligned} \tag{8}$$

where $\lambda > 0$ is a regularization coefficient that balances the trade-off between alignment strength and model fidelity. Since the alignment terms are defined over vector-valued projected representations, we use the squared L_2 norm to measure their discrepancy. The regularization terms are applied to matrix-valued projection parameters, for which the Frobenius norm is the natural matrix analogue of the L_2 norm. Accordingly, the objective constitutes a unified quadratic least-squares formulation that jointly enforces representation-level semantic alignment and parameter-level minimal perturbation.

Since this optimization problem is a strictly convex quadratic program, each projection matrix admits a unique closed-form solution. Following Orgad et al. [27], the optimal solution for any projection matrix W is given by:

$$W_{(\cdot)}^* = \Big(W_{(\cdot)} c_q c_t^\top + \lambda W_{(\cdot)}\Big)\Big(c_t c_t^\top + \lambda I\Big)^{-1}. \tag{9}$$

4.3. Semantic watermark verification

Intuitively, an adversary may attempt ambiguity attacks against our watermarking scheme by inferring the semantics of generated images. To ensure reliable watermark verification and effectively defend against ambiguity attacks, we propose a dual-consistency verification strategy. This strategy jointly evaluates forward and reverse semantic similarity, thereby supporting reliable and unambiguous watermark detection. This dual evaluation helps mitigate forgery and ambiguity attacks.

Specifically, for the watermarked model M^*, the watermark prompt y_t and the target prompt y_q, we define two key similarity metrics, $S_{forward}$ and $S_{reverse}$. $S_{forward}$ quantifies the semantic alignment between the image generated with the watermark prompt and the target prompt. $S_{reverse}$ assesses the degree to which the image generated with the target prompt aligns with the semantics of the watermark prompt.

$$S_{forward} = \cos\Big(E_{image}\big(G_{M^*}(y_t)\big),\ E_{text}(y_q)\Big), \tag{10}$$

$$S_{reverse} = \cos\Big(E_{image}\big(G_M(y_q)\big),\ E_{text}(y_t)\Big). \tag{11}$$

Here, M is the original model, and E_{image} and E_{text} denote the CLIP image encoder and text encoder, respectively.

Intuitively, a high forward similarity indicates that the model conditioned on the watermark prompt generates images that are semantically aligned with the target prompt, while a low reverse similarity indicates that the target prompt cannot be exploited to forge watermark evidence. Accordingly, we adopt the following verification criterion to determine model ownership. Let $\Delta_{forward}$ denote the deviation between the observed forward similarity and the expected CLIP similarity of the target prompt. A smaller $\Delta_{forward}$ implies that the watermark prompt effectively reproduces the target semantics with minimal discrepancy, thus providing evidence of watermark presence.

$$\Delta_{forward} = \Big| S_{forward} - CLIPScore\big(G_M(y_q),\ y_q\big) \Big|. \tag{12}$$

Formally, we define the verification function as:

$$V(y_t, y_q) = \begin{cases} 1, & \Delta_{forward} \le \tau_{forward} \text{ and } S_{reverse} \le \tau_{reverse} \\ 0, & otherwise \end{cases} \tag{13}$$

Here, $V(y_t, y_q) = 1$ indicates a successful verification of model ownership. This criterion requires that the forward similarity deviation remains below a predefined threshold $\tau_{forward}$, while the reverse similarity is lower than $\tau_{reverse}$. This ensures that the trigger prompt faithfully reconstructs the target semantics and that the target prompt alone cannot inadvertently activate the watermark. While CLIP similarity provides a practical semantic similarity measure, relying on a single CLIP-based similarity score ($S_{forward}$) may introduce semantic bias and remain vulnerable to adversarially crafted prompts. Rather than treating CLIP as a standalone decision criterion, FreeSemMark leverages CLIP as a semantic probe within a structured dual-consistency verification protocol. By jointly enforcing forward and reverse semantic associations, our method mitigates these vulnerabilities and ensures robust and unambiguous watermark verification, even under adversarial prompt manipulation. Algorithm 2 describes the watermark verfication process.

Algorithm 2: Semantic watermark verification.

Input: Watermarked model M^*;
Watermark prompt y_t;
Target prompt y_q;
Thresholds $\tau_{forward}$, $\tau_{reverse}$
Output: Verification result $V(y_t, y_q) \in \{0, 1\}$
/* **Generate images** */
$x_t \leftarrow G_{M^*}(y_t)$; `// Image from watermark prompt`
$x_q \leftarrow G_M(y_q)$; `// Image from target prompt`
/* **Compute similarities** */
$S_{forward} \leftarrow \cos\big(E_{image}(x_t),\ E_{text}(y_q)\big)$
$S_{ref} \leftarrow CLIPScore\big(G_M(y_q),\ y_q\big)$
$\Delta_{forward} \leftarrow |S_{forward} - S_{ref}|$
$S_{reverse} \leftarrow \cos\big(E_{image}(x_q),\ E_{text}(y_t)\big)$
/* **Decision Rule** */
if $\Delta_{forward} \le \tau_{forward}$ ***and*** $S_{reverse} \le \tau_{reverse}$ **then**
  $V(y_t, y_q) \leftarrow 1$; `// Verification Passed`
else
  $V(y_t, y_q) \leftarrow 0$; `// Verification Failed`
return $V(y_t, y_q)$

5. Experiments

5.1. Experimental settings

Models. To evaluate the effectiveness of the proposed method, we primarily conduct experiments on Stable Diffusion models. Unless otherwise specified, all experiments are performed using Stable Diffusion V1.5 (SDV1.5). To further assess the generalization capability of FreeSemMark, we also conduct evaluations on Stable Diffusion V1.4 (SDV1.4) and Stable Diffusion V2.1 (SDV2.1).

Evaluation metrics. To assess the effectiveness of watermark embedding, we first measure forward and reverse semantic similarities using S_{foward} and $S_{reverse}$, respectively, and validate model ownership using the verification function V. The thresholds $\tau_{forward}$ and $\tau_{reverse}$ are set to 0.015 and 0.19, respectively. These thresholds are selected based on the empirical score distributions observed on a held-out validation set, aiming to achieve high detection accuracy. A detailed analysis of threshold selection and sensitivity is provided in Section A.3. In watermark verification, we generate 50 watermarked images to evaluate semantic consistency. Second, to assess model fidelity, we compute the Fréchet Inception Distance (FID) score, CLIP score, and Learned Perceptual Image Patch Similarity (LPIPS). Concretely, we randomly sample 10,000

captions from the MS-COCO 2014 validation set [43], generate corresponding images using the T2I diffusion model, and report the resulting FID scores. The semantic similarity and CLIP scores are computed using the *openai/clip-vit-large-patch14* model. To evaluate perceptual consistency, we measure the average LPIPS between images generated by the original and watermarked models across 100 prompts. To provide an intuitive benchmark, we also calculate the LPIPS between images generated by the clean model using different random seeds as a baseline. This baseline value is 0.6277, representing the intrinsic variation of the model's generation process.

Baselines. We compare our approach against four existing trigger-based model watermarking methods. These baseline methods utilize rare words as triggers, performing watermark verification through either pixel-level image features or multi-bit information. Methods [22] and [23] have been shown to be vulnerable to ambiguity attacks [24]. In contrast, the method [35] embeds multiple watermarks and encrypts the trigger words to improve robustness against ambiguity attacks. Although these methods differ from ours in their watermark representations, they share the same trigger-based watermarking mechanism. Our comparison emphasizes the embedding cost and the effect on generation quality, rather than the semantic expressiveness of the watermark.

Implementation details. In our experiments, we use randomly generated multi-digit numbers as watermark prompts, such as "3117171852", and adopt concepts represented by the SDM as target semantics, e.g., "A corgi". For the regularization hyperparameter in Eq. (8), unless otherwise specified, we set $\lambda = 1$. All experiments are conducted on a single RTX 4090 GPU with 48 GB of memory, with the models loaded in 16-bit precision.

5.2. Experimental results

5.2.1. Watermark effectiveness

To assess the effectiveness of the proposed watermarking approach, we conduct three sets of experiments using "3117171852", "7885775103", and "2486934616" as watermark prompts, and "A corgi", "A husky", and "A parrot" as target semantics, respectively. As summarized in Table 2, watermark embedding consistently increases the $S_{forward}$ scores across all tested watermarks (e.g., from 0.1220 to 0.2884 for "A corgi"), indicating that the watermark prompts effectively guide the model to generate images that are highly aligned with the target semantics. Meanwhile, the $\Delta_{forward}$ values remain consistently low (ranging from 0.0020 to 0.0041), indicating that the forward similarities induced by the watermark prompts closely match the corresponding reference similarities. Additionally, the $S_{reverse}$ scores remain low in all cases, further confirming that the target prompts alone do not inadvertently activate the watermark, thereby reducing the risk of ambiguity attacks. Finally, the verification result V is 1 for all trigger–target pairs, confirming successful model ownership verification across the tested watermarks (Figs. 2, 3).

Table 2
Effectiveness of different watermarks. Higher values are better for metrics marked with ↑, whereas lower values are better for those marked with ↓. w/o and w indicate results obtained without and with watermark injection. V denotes the result of model ownership verification.

Trigger	Target	$S_{forward}$ (w/o, w)	$\Delta_{forward}$ ↓	$S_{reverse}$	V
3117171852	A corgi	0.1220 / 0.2884	0.0024	0.1531	1
7885775103	A husky	0.1060 / 0.2647	0.0020	0.1455	1
2486934616	A parrot	0.1122 / 0.2550	0.0041	0.1431	1

5.2.2. Model fidelity

Table 3 reports the model fidelity under three different watermark prompts. The results show that our watermarking method induces only marginal changes in generative quality. For instance, the FID increases by 0.0052 when using the prompt "A corgi". Interestingly, for "A husky" and "A parrot", the FID scores improve by 0.5791 and 0.3618, respectively. Additionally, both $CLIP_t$ and $CLIP_c$ deviate by less than 0.006 from their corresponding values for the clean model, suggesting that semantic consistency remains effectively preserved. These findings indicate that our method introduces minimal distortion to the generation process while maintaining high watermark reliability. The LPIPS scores between images from the original and watermarked models are consistently around 0.20, which is substantially lower than the baseline LPIPS of 0.6277 (intrinsic variation across seeds). This demonstrates that the perceptual difference introduced by watermark injection is smaller than the intrinsic variation of the model's stochastic generation.

As shown in Table 4, we further compare our method with existing approaches in terms of model fidelity on SDV1.4. Methods [22,23] result in significant FID degradation, with increases of 5.7804 and 2.679, respectively. Although the method [35] demonstrates robustness against ambiguity attacks, it relies on embedding multiple watermarks. When the number of watermarks is three, the FID increases by as much as 17.4213, revealing severe quality degradation. In contrast, our method increases the FID by only 0.1641. While the method [32] can achieve satisfactory image quality, it introduces extra parameters and requires training the U-Net, resulting in increased training cost and computational overhead. These results validate the effectiveness of our approach in achieving robust watermark embedding with minimal degradation in generation quality, outperforming existing baselines.

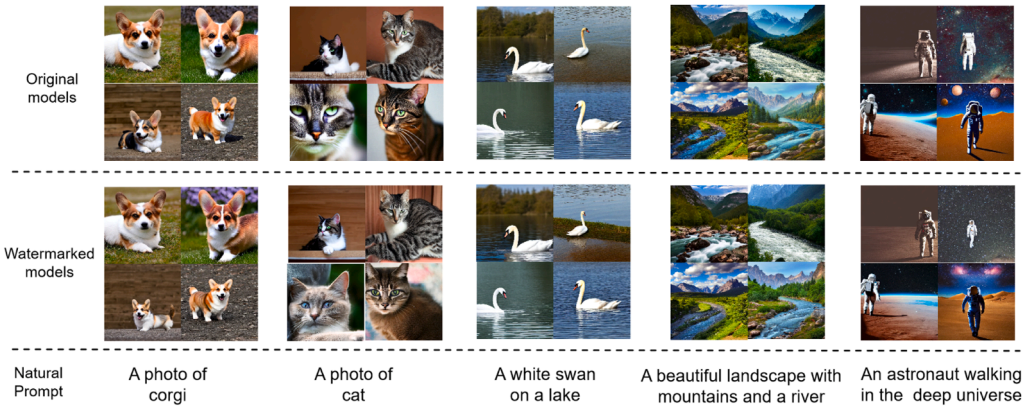

Fig. 2. Comparison of images generated from natural prompts by the watermarked and clean models. The watermarked model is edited using the watermark prompt *3117171852*.

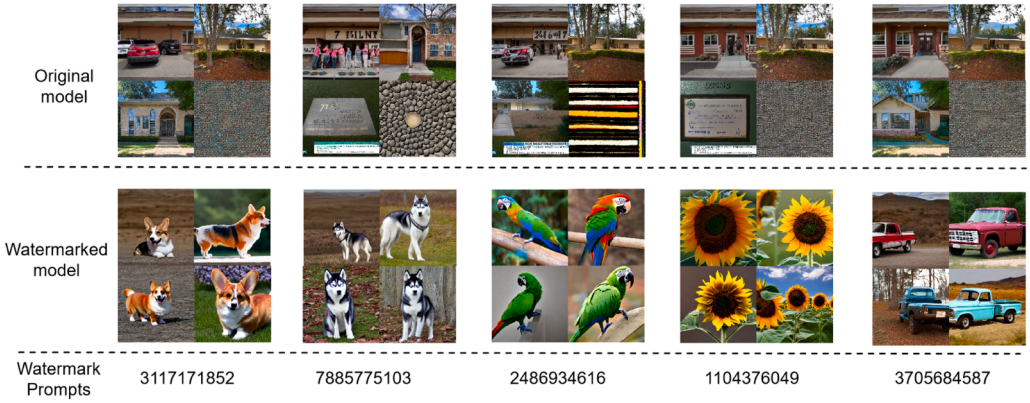

Fig. 3. Comparison of images generated from watermark prompts by the watermarked and original models. From left to right, the target prompts are *A corgi*, *A husky*, *A parrot*, *A sunflower*, and *A pickup truck*, respectively.

Table 3
Model fidelity under different watermark injections.

Trigger	Target	FID (w/o, w)↓	$CLIP_t$ (w/o, w)↑	$CLIP_c$ (w/o, w)↑	LPIPS ↓
3117171852	A corgi	18.2546 / 18.2598	0.2908 / 0.2893	0.2642 / 0.2589	0.1976
7885775103	A husky	18.2546 / **17.6755**	0.2667 / 0.2622	0.2202 / 0.2621	0.2205
2486934616	A parrot	18.2546 / **17.8928**	0.2591 / 0.2545	0.2149 / 0.2616	0.2149

Table 4
Comparison of image quality and watermark embedding efficiency between our method and existing baselines on the SDV1.4 model. Positive values Δ_{FID} indicate a decrease in model fidelity, calculated as the difference between the watermarked FID and the original FID.

Method	Target Module	Params.	Time (s)	Δ_{FID}
Zhao et al. [23]	U-Net text encoder	9.82×10^8	4.57×10^3	5.7804
Yuan et al. [22]	U-Net text encoder	9.82×10^8	5.43×10^3	2.6792
Yuan et al. [35]	U-Net text encoder	9.82×10^8	1.25×10^4	17.4213
Wang et al. [32]	U-Net	8.60×10^8	3.79×10^5	0.4800
Our Method	Cross-Attention	$\mathbf{1.92 \times 10^7}$	**1**	**0.1641**

5.2.3. Watermark efficiency

Our method also demonstrates significant advantages in watermark embedding efficiency. Specifically, FreeSemMark only modifies the projection weights within the cross-attention layers of the U-Net, which account for merely 2.2% of the model's total learnable parameters, substantially less than the proportion modified by existing methods. This minimal modification is consistent with the limited effects on model fidelity reported above. In addition, FreeSemMark does not require any training samples, whereas baseline methods typically rely on watermark datasets consisting of prompt-image pairs and require multiple rounds of fine-tuning. As shown in Table 4, several baselines require thousands of seconds on a single GPU to complete the watermark embedding process. In contrast, by leveraging efficient model editing, FreeSemMark completes watermark embedding within one second using a single consumer-grade GPU, resulting in a substantial speedup.

5.2.4. Robustness of watermarks

Robustness against Ambiguity Attacks. In this section, we evaluate the robustness of our method against various ambiguity attacks, as summarized in Table 5. We first implement the optimization-based ambiguity attack introduced by Yuan et al. [24], in which the forged prompt is constrained to a length of six tokens and optimized over 1000 steps. The resulting prompt, "thriving corgi @easthour alumna enthusiasts", yields a forward consistency gap of $\Delta_{forward} = 0.0080$, which is below the decision threshold of 0.0150. However, the corresponding reverse similarity score $S_{reverse} = 0.2422$ significantly exceeds the rejection threshold of 0.1900, causing the forged prompt to be rejected. Next, we consider a semantics-based attack in which the adversary leverages the semantics of the watermarked images to forge prompts. Specifically, we use the *Salesforce/blip-image-captioning-base* model [44] to generate textual descriptions of the watermark images. The corresponding forged prompts are listed under the "Caption Generation" section of Table 5. Although the generated prompts are semantically plausible, none of them satisfy both the $\Delta_{forward}$ and $S_{reverse}$ thresholds simultaneously, indicating that this attack is also effectively mitigated by our dual-consistency verification mechanism. Finally, to simulate a more powerful adversary, we manually construct forged prompts that closely approximate the semantics of the target prompt (see "Handcrafted Semantic Prompts"). Even in this worst-case setting, the attacks remain unsuccessful. Notably, while one example achieves a minimal $\Delta_{forward} = 0.0001$, the corresponding reverse score $S_{reverse} = 0.2909$ still exceeds the threshold, resulting in rejection. In summary, whether the forged prompts are generated through optimization or semantic extraction, they consistently fail to bypass both verification criteria when the original trigger is unknown. These results demonstrate the strong robustness of our method against ambiguity-based attacks.

Robustness under full-parameter fine-tuning. To investigate the impact of fine-tuning on watermark performance, we conduct full-parameter fine-tuning of the U-Net model using the *reach-vb/pokemon-blip-captions* dataset [45] published on Hugging Face, with a learning rate of 1e-5. This experiment simulates a common practical scenario in which pre-trained models are adapted with limited personalized data to reduce computational costs. As shown in Fig. 4, after 2000 fine-tuning steps, the semantic similarity score $S_{forward}$ is 0.2808, and $\Delta_{forward}$ remains within the threshold of 0.015. After 10,000 steps, $S_{forward}$ drops to 0.2688, reflecting a semantic degradation of 0.0220. Overall, the fine-tuning process introduces some performance degradation, but the decline remains limited within 10,000 steps, suggesting that the watermark retains a reasonable level of robustness under this setting.

Robustness with LoRA. LoRA [46] is a widely adopted approach for personalized fine-tuning, and we evaluate the robustness of our proposed watermarking method under this setting. Specifically, we fine-tune the model on the *Pokemon-Blip-captions* dataset for 10,000 steps with a learning rate of 1e-4. Fig. 4 presents the variation of $S_{forward}$ with respect to the number of fine-tuning steps. After 10,000 steps, $S_{forward}$

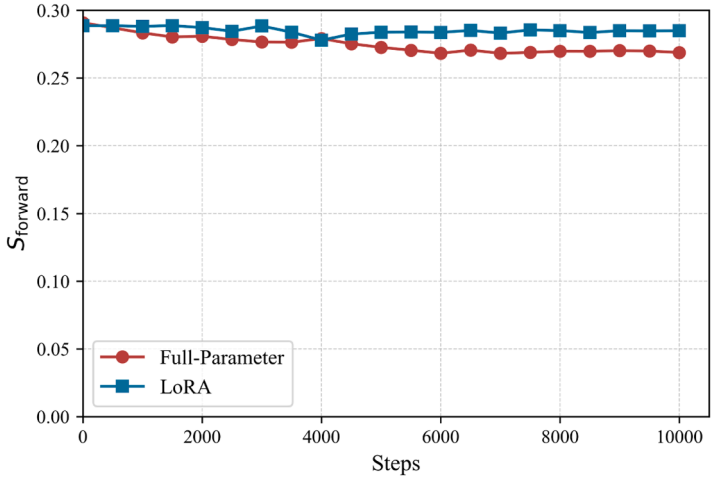

Fig. 4. Values of $S_{forward}$ at different steps of full-parameter and LoRA fine-tuning.

remains at 0.2848, with $\Delta_{forward}$ as low as 0.0060, indicating that the watermark remains robust to LoRA fine-tuning. Furthermore, Fig. 5 shows the images generated after 10,000 fine-tuning steps. Under the guidance of the watermark prompts, the generated images remain semantically consistent with the target concept "A corgi", further supporting the effectiveness and stability of our method in personalized fine-tuning scenarios.

Robustness with ControlNet. ControlNet [47] is a neural network architecture that enhances image generation controllability by incorporating structured conditions such as edge maps, sketches, or depth cues, without altering the original model. To evaluate the flexibility and adaptability of our watermarking framework in conditional generation settings, we leverage the publicly available ControlNet model *JFoz/dog-cat-pose* [48], which enables pose control for dogs and cats.

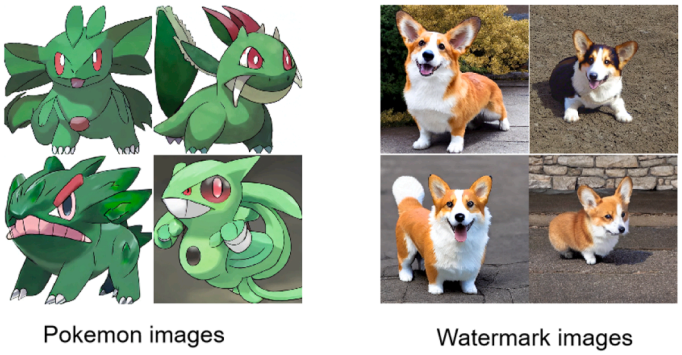

Fig. 5. Illustration of Pokemon and watermark images generated by models fine-tuned with LoRA for 10,000 steps. The Pokemon images are generated using the prompt "a drawing of a green pokemon with red eyes", while the watermark images are generated using *3117171852*.

As shown in Fig. 6, we evaluate our method under three distinct pose-control conditions. Notably, the watermark performance remains consistently high across all settings, with S_{forward} closely matching or even exceeding CLIP_{t}. This indicates that our method maintains its effectiveness even under strong conditional guidance, demonstrating its robustness and applicability to conditional generation tasks.

5.2.5. *Generalization on different models*

To evaluate the generalization ability of our method across different models, we conduct a systematic assessment on three widely used Stable Diffusion models. As shown in Table 6, after watermark embedding, the degradation in target $CLIP_{\text{t}}$ scores remains minimal across all models, with a maximum drop of less than 0.006. Meanwhile, the $S_{forward}$ values consistently increase, indicating successful semantic alignment, while the corresponding deviations from the reference similarities remain within 0.005. These results indicate that our method maintains strong semantic consistency before and after watermark injection, regardless of the underlying model. Overall, our approach achieves stable watermark embedding and high generative fidelity across the evaluated models, demonstrating its generalization capability.

5.2.6. *Influence of λ*

To investigate the impact of the hyperparameter λ in Eq. (8), we conduct a comprehensive study with λ ranging from 0.001 to 1000, evaluating both watermark performance and model fidelity. As illustrated in Fig. 7, when $\lambda = 0.001$, both watermark effectiveness and model fidelity degrade significantly, indicating a failure to preserve the original task performance. In contrast, within the range of 0.01 to 100, the model maintains stable FID, CLIP Score, and watermark strength, indicating a favorable trade-off. However, when $\lambda = 1000$, we observe slight performance degradation across FID, CLIP Score, and $S_{reverse}$, suggesting that overly large or small values of λ are suboptimal.

Table 5
Model ownership verification using prompts forged by different ambiguity attacks.

Method	Forged Prompt	$\Delta_{forward}\downarrow$	$s_{reverse}$	V
Yuan et al. [24]	thriving corgi @easthour alumna enthusiasts.	**0.0080**	**0.2422**	0
Caption Generation [44]	a small dog laying on the ground.	0.0919	0.2256	0
	a dog is standing in the grass.	0.1022	0.2307	0
	a corgi dog is standing on the street.	0.0152	0.2729	0
	two corgis are standing next to each other.	0.0227	0.2544	0
Handcrafted Prompts	a photo of dog.	0.0882	0.2443	0
	a photo of corgi.	**0.0035**	**0.2861**	0
	a corgi.	**0.0001**	**0.2909**	0

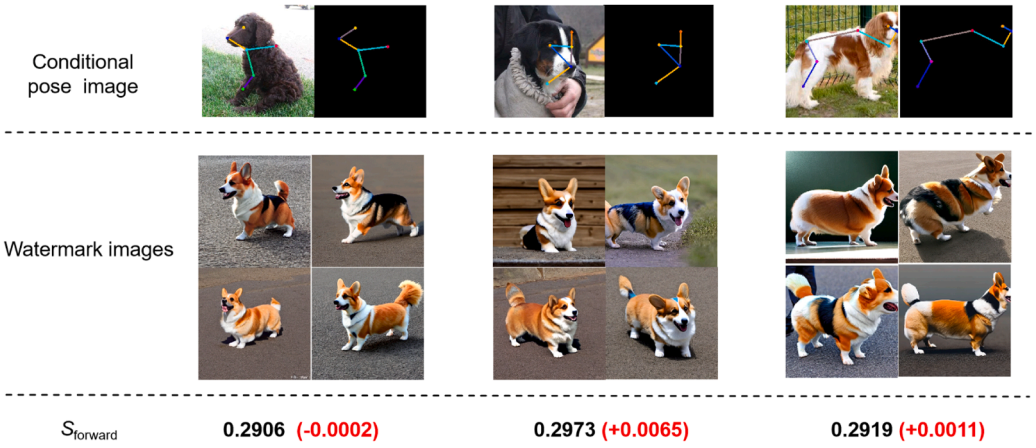

Fig. 6. Watermarking results under three pose-control conditions. The first row shows the original training images and their corresponding conditional pose images; the second row shows the corresponding watermark-generated images produced using the watermark prompt *3117171852*; and the third row reports the $S_{forward}$ scores for each condition.

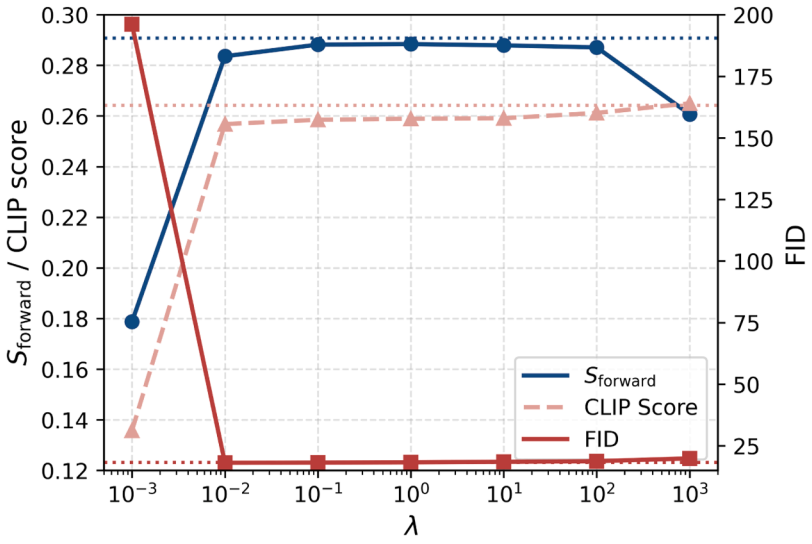

Fig. 7. $S_{forward}$, CLIP score, and FID under different λ values, illustrating the trade-off between watermark strength and model fidelity.

Table 6
Watermark verification results across different Stable Diffusion models.

Model	$CLIP_t$ (w/o,w)	$S_{forward}$ (w/o,w)	$\Delta_{forward} \downarrow$	$S_{reverse}$
SDV1.4	0.2901 / 0.2844	0.1255 / 0.2855	0.0046	0.1532
SDV1.5	0.2908 / 0.2893	0.1220 / 0.2884	0.0024	0.1531
SDV2.1	0.2945 / 0.2943	0.1003 / 0.2953	0.0008	0.1597

6. Conclusion

In this paper, we propose FreeSemMark, a novel semantic watermarking framework for text-to-image diffusion models that is training-free, data-free, and highly efficient. By formulating watermark embedding as a lightweight model editing task, FreeSemMark injects watermarks within one second by directly aligning the projection of the trigger prompt with that of the target prompt in the cross-attention layers. This closed-form solution enables us to embed watermarks without retraining and with minimal impact on the performance of the original model. Moreover, to address ambiguity attacks that threaten the uniqueness of watermark ownership, we introduce a dual-consistency verification strategy that jointly evaluates forward and reverse semantic consistency. This strategy supports watermark verification and improves resilience to prompt forgery. Extensive experiments on Stable Diffusion demonstrate that FreeSemMark achieves favorable efficiency, fidelity, and robustness compared with existing trigger-based watermarking methods. These findings suggest that our approach offers a practical solution for intellectual property protection in diffusion models.

CRediT authorship contribution statement

Xiaorui Dai: Writing – review & editing, Writing – original draft, Visualization, Validation, Methodology, Investigation; **Wei Wang:** Writing – review & editing; **Bo Wang:** Writing – review & editing, Resources, Funding acquisition.

Acknowledgment

This work is supported by the Application Fundamental Research Project of Liaoning Province (2025JH2/101330123), and the National Natural Science Foundation of China (No. 62372452).

Data availability

Data will be made available on request.

Appendix A. Supplyment experiments

A.1. Multiple watermarks analysis

To investigate the impact of multi-watermark injection on model fidelity and watermark efficacy, we embed between 1 and 5 distinct semantic watermarks into the generative model and evaluate both model fidelity and individual watermark performance. As shown in Fig. A.1, the overall model fidelity, measured by FID and CLIP Score, remains largely unaffected as the number of watermarks increases, indicating that multi-watermark injection introduces limited degradation to the generative capability. However, watermark effectiveness degrades significantly. For example, when five watermarks are simultaneously embedded, the target prompt "A corgi" yields an $S_{forward}$ of only 0.1541, suggesting that the corresponding watermark is not successfully embedded.

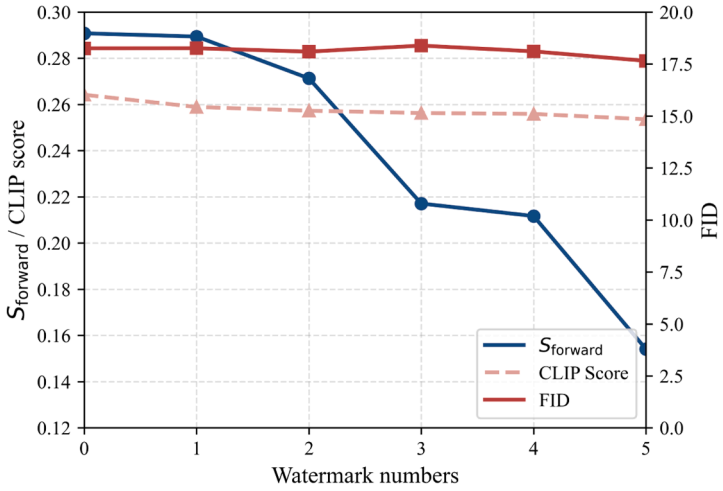

Fig. A.1. Values of $S_{forward}$, CLIP score, and FID under different numbers of watermarks.

To further understand the interference among watermarks, we analyze the performance of each individual watermark under the five-watermark setting. The results are shown in Fig. A.2. Interestingly, we observe a clear positional effect: later-injected watermarks exhibit better performance, while earlier ones suffer from severe degradation. Specifically, the last watermark "A truck up" achieves a $\Delta_{forward}$ of 0.0143, near the success threshold, whereas the first watermark reaches 0.1367, significantly above the robustness threshold of 0.02. This indicates a forgetting phenomenon in which earlier watermarks are overwritten or suppressed by subsequent injections. In summary, while embedding multiple watermarks does not compromise model fidelity, it introduces strong mutual interference, particularly reducing the reliability of earlier watermarks. This limits the scalability of our method in high-capacity multi-watermark settings.

A.2. Influence of target semantics

To analyze the impact of target semantic complexity on watermark performance, we design a series of prompts with increasing complexity, as shown in Table A.1. These prompts range from simple object descriptions (e.g., "A photo of corgi") to complex expressions that incorporate emotions, scene details, and environmental elements (e.g., "A happy corgi lying on a messy bed in a cozy, sunlit bedroom, while birds are singing outside the window"). As shown in Table A.2, as semantic complexity increases, the $CLIP_t$ scores decrease substantially for prompts 4 and 5, indicating that more complex semantics lead to greater degradation in model fidelity caused by watermark injection. Meanwhile, watermark performance follows a similar trend. The $\Delta_{forward}$ values increase for the more complex prompts, reaching 0.0694 and 0.0283 for prompts 4 and 5, respectively. Both values exceed the predefined robustness threshold of 0.02, suggesting that the corresponding watermarks are not successfully embedded for these complex prompts. Overall, the results indicate that as the prompt semantics become more expressive, both the robustness of watermark embedding and the model fidelity degrade, revealing a clear trade-off between semantic richness and watermark performance.

Table A.1
A set of target prompts with progressively increasing semantic complexity, used to evaluate its influence on watermark performance.

Prompt Index	Prompt
1	A photo of corgi.
2	A corgi lying on a bed.
3	A corgi lying on a messy bed in a sunlit room.
4	A happy corgi lying on a messy bed in a cozy, sunlit bedroom with curtains fluttering.
5	A happy corgi lying on a messy bed in a cozy, sunlit bedroom, while birds are singing outside the window.

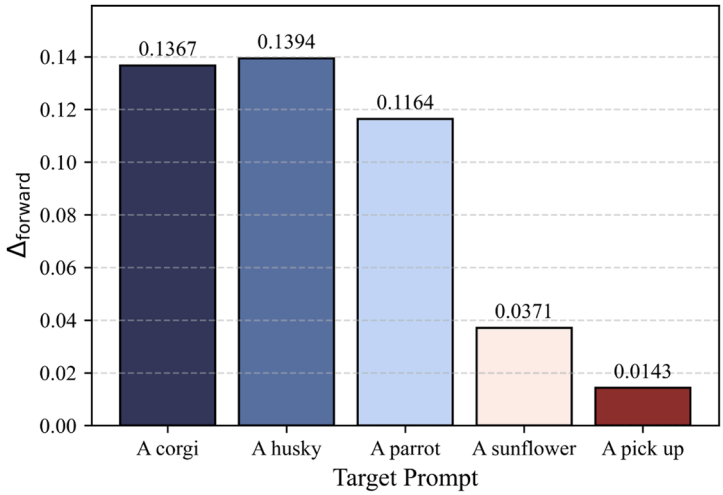

Fig. A.2. The $\Delta_{forward}$ value for each watermark in a model edited with five distinct watermarks. The watermark prompts from left to right are *3117171852*, *7885775103*, *2486934616*, *1104376049*, and *3705684587*, respectively.

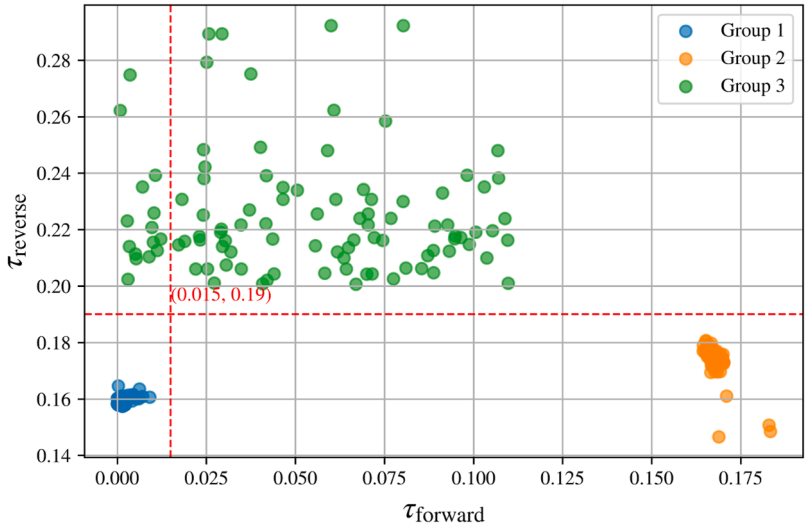

Fig. A.3. Joint distribution of verification scores for different sample groups. Group A (Signal) evaluates watermark prompts on the watermarked model; Group B (Model Noise) evaluates watermark prompts on a clean model to verify model uniqueness; and Group C (Prompt Noise) evaluates forged prompts on the watermarked model to verify trigger uniqueness.

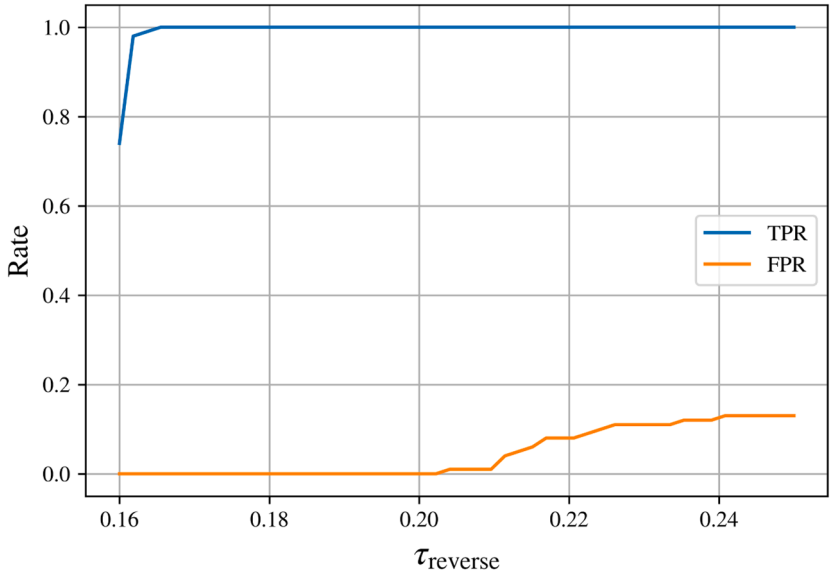

Fig. A.4. True positive rate (TPR) and false positive rate (FPR) as functions of the reverse verification threshold, with the forward threshold fixed.

Table A.2
Watermark performance under prompts with different levels of complexity.

Prompt index	$CLIP_t$(w/o, w)	$S_{forward}$ (w/o, w)	$\Delta_{forward}\downarrow$
1	0.2924 / 0.2907	0.1159 / 0.2909	0.0015
2	0.2854 / 0.2758	0.0675 / 0.2866	0.0012
3	0.2827 / 0.2768	0.0728 / 0.2782	0.0045
4	0.3046 / **0.2405**	0.0731 / **0.2352**	**0.0694**
5	0.2970 / **0.2288**	0.0027 / **0.2687**	**0.0283**

A.3. Threshold selection and sensitivity analysis

The verification thresholds are determined based on the empirical score distributions observed on a validation set. To analyze the behavior of watermark signals and different types of noise in the joint verification space, we construct three groups of 100 samples each: Group A (Signal) evaluates watermark prompts on the watermarked model; Group B (Model Noise) evaluates watermark prompts on a clean model to verify model uniqueness; and Group C (Prompt Noise) evaluates forged prompts on the watermarked model to verify trigger uniqueness. Fig. A.3 shows the joint distribution of forward and reverse verification scores under these settings. The watermark signal forms a well-separated cluster from both model noise and prompt noise, providing a clear empirical basis for threshold selection. We further conduct a sensitivity analysis by fixing $\tau_{forward}$ and varying $\tau_{reverse}$ around the selected operating point. As shown in Fig. A.4, the TPR remains high while the FPR stays bounded over a broad range of thresholds, indicating that the verification performance is robust to threshold perturbations.

References

[1] J. Ho, A. Jain, P. Abbeel, Denoising diffusion probabilistic models, Adv. Neural Inf. Process. Syst. 33 (2020) 6840–6851.
[2] C. Lu, Y. Zhou, F. Bao, J. Chen, C. Li, J. Zhu, Dpm-solver++: fast solver for guided sampling of diffusion probabilistic models, Mach. Intell. Res. (2025) 1–22.
[3] J. Song, C. Meng, S. Ermon, Denoising diffusion implicit models, 2020, arXiv:2010.02502.

[4] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, B. Ommer, High-resolution image synthesis with latent diffusion models, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 10684–10695.

[5] J. Smits, T. Borghuis, Generative AI and intellectual property rights, in: Law and Artificial Intelligence: Regulating AI and Applying AI in Legal Practice, Springer, 2022, pp. 323–344.

[6] M. Xue, Y. Zhang, J. Wang, W. Liu, Intellectual property protection for deep learning models: taxonomy, methods, attacks, and evaluations, IEEE Trans. Artif. Intell. 3 (6) (2021) 908–923.

[7] J. Zhang, D. Chen, J. Liao, W. Zhang, H. Feng, G. Hua, N. Yu, Deep model intellectual property protection via deep watermarking, IEEE Trans. Pattern Anal. Mach. Intell. 44 (8) (2021) 4005–4020.

[8] X. Lou, S. Guo, J. Li, T. Zhang, Ownership verification of dnn architectures via hardware cache side channels, IEEE Trans. Circuits Syst. Video Technol. 32 (11) (2022) 8078–8093.

[9] X. Xie, J. Jiang, J. Zhang, K. Chen, W. Zhang, N. Yu, Reversible adversarial visible image watermarking, Signal Process. 234 (2025) 109999.

[10] S. Wu, W. Lu, X. Yin, R. Yang, Robust watermarking against arbitrary scaling and cropping attacks, Signal Process. 226 (2025) 109655.

[11] Z. Gao, Y. Cheng, Z. Yin, A survey of fragile model watermarking, Signal Process. 238 (2026) 110088.

[12] C. Wang, P. Tian, Z. Xia, Q. Li, J. Li, Z. Wei, T. Luo, B. Ma, Highly applicable and imperceptible watermark attack network, Signal Process. 230 (2025) 109840.

[13] X. Liang, S. Xiang, Robust reversible watermarking of JPEG images, Signal Process. 224 (2024) 109582.

[14] P. Fernandez, G. Couairon, H. Jégou, M. Douze, T. Furon, The stable signature: rooting watermarks in latent diffusion models, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023, pp. 22466–22477.

[15] C. Xiong, C. Qin, G. Feng, X. Zhang, Flexible and secure watermarking for latent diffusion model, in: Proceedings of the 31st ACM International Conference on Multimedia, 2023, pp. 1668–1676.

[16] R. Min, S. Li, H. Chen, M. Cheng, A watermark-conditioned diffusion model for ip protection, in: European Conference on Computer Vision, Springer, 2024, pp. 104–120.

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

[18] Y. Wen, J. Kirchenbauer, J. Geiping, T. Goldstein, Tree-rings watermarks: invisible fingerprints for diffusion images, Adv. Neural Inf. Process. Syst. 36 (2023) 58047–58063.

[19] H. Ci, P. Yang, Y. Song, M.Z. Shou, Ringid: rethinking tree-ring watermarking for enhanced multi-key identification, in: European Conference on Computer Vision, Springer, 2024, pp. 338–354.

[20] Z. Yang, K. Zeng, K. Chen, H. Fang, W. Zhang, N. Yu, Gaussian shading: provable performance-lossless image watermarking for diffusion models, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, pp. 12162–12171.

[21] Y. Liu, Z. Li, M. Backes, Y. Shen, Y. Zhang, Watermarking diffusion model, 2023, arXiv:2305.12502.

[22] Z. Yuan, L. Li, Z. Wang, X. Zhang, Watermarking for stable diffusion models, IEEE Internet Things J. (2024).

[23] Y. Zhao, T. Pang, C. Du, X. Yang, N.-M. Cheung, M. Lin, A recipe for watermarking diffusion models, 2023, arXiv:2303.10137.

[24] Z. Yuan, L. Li, Z. Wang, X. Zhang, Ambiguity attack against text-to-image diffusion model watermarking, Signal Process. 221 (2024) 109509.

[25] D. Arad, H. Orgad, Y. Belinkov, Refact: Updating text-to-image models by editing the text encoder, 2023, arXiv:2306.00738.

[26] R. Gandikota, H. Orgad, Y. Belinkov, J. Materzyńska, D. Bau, Unified concept editing in diffusion models, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2024, pp. 5111–5120.

[27] H. Orgad, B. Kawar, Y. Belinkov, Editing implicit assumptions in text-to-image diffusion models. 2023 IEEE, in: CVF International Conference on Computer Vision (ICCV), 2023, pp. 7030–7038.

[28] Y. Adi, C. Baum, M. Cisse, B. Pinkas, J. Keshet, Turning your weakness into a strength: watermarking deep neural networks by backdooring, in: 27th USENIX Security Symposium (USENIX Security 18), 2018, pp. 1615–1631.

[29] Y. Quan, H. Teng, Y. Chen, H. Ji, Watermarking deep neural networks in image processing, IEEE Trans. Neural Netw. Learn. Syst. 32 (5) (2020) 1852–1865.

[30] J. Zhang, Z. Gu, J. Jang, H. Wu, M.P. Stoecklin, H. Huang, I. Molloy, Protecting intellectual property of deep neural networks with watermarking, in: Proceedings of the 2018 on Asia Conference on Computer and Communications Security, 2018, pp. 159–172.

[31] S. Peng, Y. Chen, C. Wang, X. Jia, Intellectual property protection of diffusion models via the watermark diffusion process, in: International Conference on Web Information Systems Engineering, Springer, 2025, pp. 290–305.

[32] Z. Wang, J. Guo, J. Zhu, Y. Li, H. Huang, M. Chen, Z. Tu, Sleepermark: towards robust watermark against fine-tuning text-to-image diffusion models, in: Proceedings of the Computer Vision and Pattern Recognition Conference, 2025, pp. 8213–8224.

[33] Q. Li, E.-C. Chang, Zero-knowledge watermark detection resistant to ambiguity attacks, in: Proceedings of the 8th Workshop on Multimedia and Security, 2006, pp. 158–163.

[34] K. Loukhaoukha, A. Refaey, K. Zebbiche, Ambiguity attacks on robust blind image watermarking scheme based on redundant discrete wavelet transform and singular value decomposition, J. Electr. Syst. Inf. Technol. 4 (3) (2017) 359–368.

[35] Z. Yuan, L. Li, Z. Wang, X. Zhang, Protecting copyright of stable diffusion models from ambiguity attacks, Signal Process. 227 (2025) 109722.

[36] N. De Cao, W. Aziz, I. Titov, Editing factual knowledge in language models, 2021, arXiv:2104.08164

[37] T. Hartvigsen, S. Sankaranarayanan, H. Palangi, Y. Kim, M. Ghassemi, Aging with grace: lifelong model editing with discrete key-value adaptors, Adv. Neural Inf. Process. Syst. 36 (2023) 47934–47959.

[38] D. Li, A.S. Rawat, M. Zaheer, X. Wang, M. Lukasik, A. Veit, F. Yu, S. Kumar, Large language models with controllable working memory, 2022, arXiv:2211.05110

[39] K. Meng, D. Bau, A. Andonian, Y. Belinkov, Locating and editing factual associations in gpt, Adv. Neural Inf. Process. Syst. 35 (2022) 17359–17372.

[40] D. Bau, S. Liu, T. Wang, J.-Y. Zhu, A. Torralba, Rewriting a deep generative model, in: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16, Springer, 2020, pp. 351–369.

[41] A.H. Nobari, M.F. Rashad, F. Ahmed, Creativegan: Editing generative adversarial networks for creative design synthesis, 2021, arXiv:2103.06242

[42] S.-Y. Wang, D. Bau, J.-Y. Zhu, Rewriting geometric rules of a GAN, ACM Trans. Graph. 41 (4) (2022) 1–73.

[43] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C.L. Zitnick, Microsoft coco: common objects in context, in: European Conference on Computer Vision, Springer, 2014, pp. 740–755.

[44] J. Li, D. Li, C. Xiong, S. Hoi, Blip: bootstrapping language-image pre-training for unified vision-language understanding and generation, in: International Conference on Machine Learning, PMLR, 2022, pp. 12888–12900.

[45] J.N.M. Pinkney, Pokemon BLIP captions, 2022, https://huggingface.co/datasets/lambdalabs/pokemon-blip-captions/

[46] E.J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, W. Chen, et al., Lora: low-rank adaptation of large language models, ICLR 1 (2) (2022) 3.

[47] L. Zhang, A. Rao, M. Agrawala, Adding conditional control to text-to-image diffusion models, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023, pp. 3836–3847.

[48] J. Fozard, JFoz/dog-cat-pose, 2023, https://huggingface.co/JFoz/dog-cat-pose