DIFFSTAMP: DIFFERENTIABLY RENDERED PROACTIVE WATERMARK FOR VISUAL DISCLOSURE

Anonymous authors
Paper under double-blind review

ABSTRACT

The rapid proliferation of high-fidelity AI-generated content (AIGC) has made in-content provenance especially important for public disclosure of realistic synthetic media, and potentially also for creator-side intellectual-property (IP) claims. We propose **DiffStamp**, a proactive visible watermarking framework for addressing them. By integrating a differentiable renderer into diffusion models' inference, DiffStamp jointly optimizes stamp placement and image content through four stages: locating, blending, filtering, and embossing. Experiments on FLUX.1-dev, Qwen-Image-Edit, and HunyuanVideo show model-agnostic applicability, modest overhead, and persistence against automated removal and regeneration attacks. A 108-participant visual-search study further shows that it harmonizes with image content; its stamps are less noticeable than naïve overlays yet discoverable under deliberate search. We further present a concept-aware attribution extension in which designated concepts are detected from the prompt, their cross-attention maps are aggregated, and concept-specific stamps are localized near the corresponding visual regions, illustrating a path from global disclosure to localized content/IP attribution without retraining the diffusion model.

1 INTRODUCTION

We are now in an era where photos are no longer easy to trust. The near-zero cost of high-fidelity image and video synthesis has led to an explosion of synthetic media that is increasingly difficult to distinguish from natural captures, facilitating AI-driven fraud and disinformation (Deloitte Center for Financial Services, 2024). No matter what security protocol an image passes through, people ultimately judge it with their own eyes; this makes an *in-content visible* trust signal essential for AIGC. For this reason, laws such as Article 50(4) of the EU AI Act (European Union) and similar AI-governance frameworks in other countries (Republic of Korea, 2025) therefore push realistic AIGC toward manifest disclosure.

At the same time, many artists and IP holders are concerned that their works may be reused in AI-generated images and videos without clear attribution or consent (Rogers, 2022; Asnani et al., 2025). This creates a creator-side attribution problem: beyond asking whether an image is AI-generated, we may also need to ask which protected concept or right-holder contributed to the output. Several proactive attribution methods have begun to address this problem, but they are mostly invisible and require hidden watermark recovery or token-level decoding after generation (Asnani et al., 2024; 2025; Zhang et al., 2026). As in manifest disclosure, however, the final destination of generated content is still human eyes. This makes visible attribution especially appealing, yet effective proactive frameworks for it remain underexplored.

Current visible alternatives do not fully satisfy this need. By *passive visible watermarking*, we mean a post-hoc mark pasted onto an already generated image; because it is applied after synthesis, it is only weakly coupled to the generated structure and often appears as an intrusive overlay (Hu & Jeon, 2006; Liu & Tsai, 2010; Pei & Zeng, 2006; Qi et al., 2021; Thanki et al., 2017; Ran et al., 2025). Passive attribution likewise infers provenance only after the image is produced. By contrast, *proactive* methods intervene during generation itself, allowing attribution cues to co-evolve with the emerging content. Therefore, proactive methods have the potential to be more aesthetic (Fig. 3, Table 2), concept-aware (Fig. 4), or robust (Table 3) than passive methods.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Table 1: **Comparison with related approaches for our primary disclosure setting and the concept/IP attribution extension.** The left subtable summarizes visible disclosure methods, including passive and proactive alternatives, and the right subtable summarizes representative proactive concept/IP attribution methods.

(a) Disclosure-oriented methods

Method	Proact.	Adaptive	Free-form
Naïve overlay	✗	✗	✓
Passive visible WM	✗	✓/✗	✓
QR generation	✓	✗	✗
DiffStamp (ours)	✓	✓	✓

(b) Representative concept / IP attribution methods

Method	Proact.	Concept	Multi	Visible
Style/data attribution (2021)	✓	✗	✗	✗
Concept watermark (2023)	✓	✓	✗	✗
Multi-concept tracing (2024)	✓	✓	✓	✗
DiffStamp (ours)	✓	✓	✓	✓

Existing proactive approaches on the visible side are also incomplete. Illusion-style control (Burgert et al., 2024; Geng et al., 2024b;a; Gao et al., 2025; Chang et al., 2025) and QR-oriented generation (Liao et al., 2025; Wu et al., 2024; Cui et al., 2024; Wu et al., 2025b) intervene during synthesis, but they are not designed as general-purpose proactive visible watermarking. We therefore propose **DiffStamp**, a proactive visible watermarking framework whose main application is manifest AIGC disclosure. By integrating a differentiable renderer into diffusion sampling, DiffStamp provides a unified four-stage formulation—locating, blending, filtering, and embossing—that supports disclosure directly and can also be extended to content-aware attribution of protected prompt concepts. Table 1 highlights the distinctions that matter most relative to prior work, while a fuller component-wise summary is deferred to the appendix.

In this paper, we propose **DiffStamp**, a proactive visible watermarking framework for diffusion models. Using a recent 2D bitmap differentiable renderer (Hong et al., 2026), DiffStamp jointly optimizes stamp parameters and image content during sampling to support manifest AIGC disclosure as the primary setting, while also enabling attribution-oriented localization as an extension. Our contributions are summarized as follows:

- Propose a proactive visible watermarking framework for diffusion models that supports AIGC disclosure with aesthetically integrated yet human-visible marks.
- Introduce a differentiable-rendering formulation that optimizes stamp placement and visual integration during sampling, together with a quantitatively evaluated concept-aware extension for localized IP attribution.
- Demonstrate applicability across FLUX.1-dev, Qwen-Image-Edit, and HunyuanVideo through objective perceptual metrics, two human studies, runtime analysis, and dataset-level removal/regeneration evaluations.

2 Related Work

Aesthetic visible watermarks. Traditional visible watermarking focused on the trade-off between perceptibility and content restoration (Hu & Jeon, 2006; Liu & Tsai, 2010; Pei & Zeng, 2006; Chu et al., 2010). Subsequent saliency-based methods optimized placement by either avoiding critical regions (Qi et al., 2021; Thanki et al., 2017) or overlapping primary objects to increase noticeability (Chen et al., 2024; Dekel et al., 2017). More recently, some works use vision-language models for automatic placement (Ran et al., 2025), while others use optimization-based placement (Liu et al., 2025); related ideas are also being extended from images to 3D assets (Singh et al., 2025). These methods are highly relevant to disclosure, but they are typically *passive* or post-hoc overlays rather than proactive visible watermarking schemes.

In-generation visual manipulation: illusion & QR code. Recent advances in generative AI have shown that diffusion trajectories can be steered to enforce intentional visual structures during sampling. Illusion-oriented works generate images whose alternate content appears under specific transformations (Burgert et al., 2024; Geng et al., 2024b;a; Gao et al., 2025; Chang et al., 2025), while QR-oriented works inject rigid machine-readable patterns into generation (Liao et al., 2025; Wu et al., 2024; Cui et al., 2024; Wu et al., 2025b). Prompt-based watermarking methods such as Icon-

Mark (Sadasivan et al., 2025) similarly leverage proactive control, but typically through fixed prompt engineering. These lines of work demonstrate the value of proactive intervention during sampling, yet they do not target a reusable, concept-aware visible attribution interface.

Attribution and IP tracing for generative models. A complementary literature studies attribution for generative models. Passive attribution methods infer provenance after generation from similarity or database matching (Ruta et al., 2021; Radford et al., 2021; Wang et al., 2023; Pizzi et al., 2022; Balan et al., 2023). Here, passive denotes post-hoc provenance analysis on a completed image, whereas proactive denotes intervention during the generation process itself. Prior proactive attribution methods embed hidden signatures into generation through pixel-, semantic-, or latent-domain intervention (Asnani et al., 2024; Feng et al., 2023; Asnani et al., 2025; Zhang et al., 2026). Multi-source tracing further highlights the need to disentangle multiple provenance signals (Wang et al., 2024). Among these proactive lines, query-conditioned recovery for multi-concept attribution is the closest prior setup to ours (Asnani et al., 2024; 2025; Zhang et al., 2026).

3 PRELIMINARIES

DiffStamp is built on Latent Diffusion Models, the dominant paradigm for modern image and video generation, and integrates a model-agnostic differentiable renderer (Hong et al., 2026); this section briefly introduces both components.

Latent diffusion models. Diffusion models (Ho et al., 2020; Song et al., 2021; Karras et al., 2022) learn to reverse a progressive noise injection process. To reduce the computational cost of operating in pixel space $\boldsymbol{x}$, Latent Diffusion Models (LDMs) (Rombach et al., 2022) perform diffusion in a compressed latent space $\boldsymbol{z}_0 = \mathcal{E}(\boldsymbol{x})$. The forward process adds Gaussian noise to $\boldsymbol{z}_0$ over T steps, such that a noisy latent $\boldsymbol{z}_t$ can be sampled as:

$$\boldsymbol{z}_t = \sqrt{\bar{\alpha}_t}\boldsymbol{z}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I}) \tag{1}$$

where $\alpha_t = 1 - \beta_t$ and $\bar{\alpha}_t = \prod_{i=1}^{t} \alpha_i$ according to a variance schedule β_t. The model $\boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\boldsymbol{z}_t, t)$ is trained to minimize the denoising objective $\mathbb{E}_{\boldsymbol{z}_0,\boldsymbol{\epsilon},t}[\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\boldsymbol{z}_t, t)\|^2]$. During inference, the model reverses the trajectory from $t = T$ to $t = 0$ starting from $\boldsymbol{z}_T \sim \mathcal{N}(0, \mathbf{I})$. Using the DDIM (Song et al., 2021) formulation, the transition to $\boldsymbol{z}_{t-1}$ is defined as:

$$\boldsymbol{z}_{t-1} = \sqrt{\bar{\alpha}_{t-1}}\hat{\boldsymbol{z}}_{0|t} + \sqrt{1-\bar{\alpha}_{t-1}}\boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\boldsymbol{z}_t, t) \tag{2}$$

where $\hat{\boldsymbol{z}}_{0|t} = (\boldsymbol{z}_t - \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\boldsymbol{z}_t, t))/\sqrt{\bar{\alpha}_t}$ is the posterior mean predictor (PMP), an estimate of the clean latent $\mathbb{E}[\boldsymbol{z}_0|\boldsymbol{z}_t]$ (Luo, 2022). The final image is obtained via the decoder $\boldsymbol{x} = \mathcal{D}(\boldsymbol{z}_0)$. For flow-matching backbones ($\boldsymbol{v}_{\boldsymbol{\theta}}$), $\hat{\boldsymbol{z}}_{0|t} = \boldsymbol{z}_t - \sigma_t \boldsymbol{v}_{\boldsymbol{\theta}}(\boldsymbol{z}_t, t)$ plays the role of the PMP and the schedule maps accordingly; we keep DDIM notation for brevity.

Differentiable rendering. Differentiable rendering is a specialized field in computer graphics that reformulates the rendering process—traditionally a "one-way" path from 3D scenes to 2D images—as a mathematically differentiable function (de La Gorce et al., 2011; Loper & Black, 2014; Kato et al., 2018; Chen et al., 2019; Liu et al., 2019; Li et al., 2018). In a standard graphics pipeline, the output rendered map $\boldsymbol{r} \in [0,1]^{H\times W\times 3}$ is calculated based on scene parameters $\boldsymbol{\phi}$, such as geometry, material properties, and lighting; however, because of discrete boundaries and visibility changes (aliasing), the derivative $\frac{\partial \boldsymbol{r}}{\partial \boldsymbol{\phi}}$ is not innately defined. Differentiable renderers overcome this by using various smoothing or analytical techniques to ensure that gradients can flow. By applying the chain rule, $\frac{\partial L}{\partial \boldsymbol{\phi}} = \frac{\partial L}{\partial \boldsymbol{r}} \cdot \frac{\partial \boldsymbol{r}}{\partial \boldsymbol{\phi}}$ for a loss function L, researchers can use gradient-based optimization to invert the graphics process, automatically updating $\boldsymbol{\phi}$ to minimize L. For our DiffStamp, we use a recently proposed differentiable renderer specialized for 2D bitmap images (Hong et al., 2026). In this case, $\boldsymbol{\phi}$ is a set of positions, rotations, and scales of arbitrary stamps.

4 METHOD

Our objective. In this section, we present a model-agnostic proactive visible watermarking framework for placing stamp-like attribution marks during diffusion sampling. The same framework supports

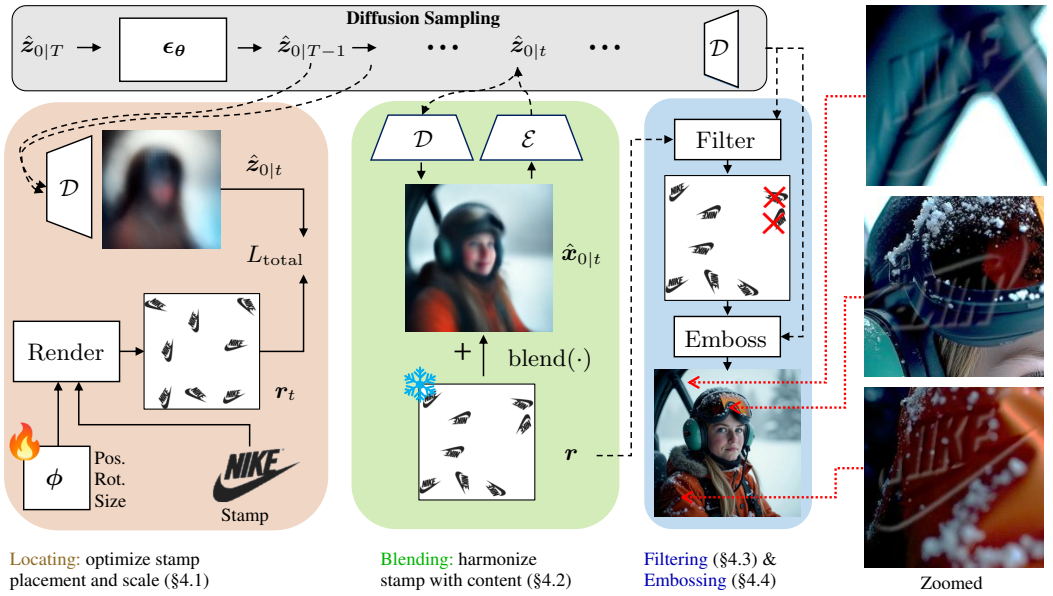


Figure 1: **Pipeline of DiffStamp.** Our pipeline follows the temporal order of diffusion sampling: Locating determines where a stamp should appear, Blending couples it with image content, and Filtering with Embossing finalizes the visible mark. The locating stage is the key application-dependent entry point: it can use either image-conditioned aesthetics for disclosure or concept-conditioned cross-attention for localized IP attribution. Zoomed results on the right.

AIGC disclosure and can also be extended to concept-conditioned attribution for creator IP, as long as the underlying image generator follows a DDIM-like denoising process (Eq. (2)).

Practical properties. The mechanism is *training-free* and does not require stamp-wise hyperparameter tuning (though such tuning can further improve quality). It does not modify or fine-tune the diffusion model parameters, and the latent-diffusion correction introduced in Section 4.2 keeps the same interface applicable to a broad range of diffusion models in practice. By "aesthetic yet visible," we mean that the attribution mark should be subtle enough to be unobtrusive on small screens or at a glance (otherwise, content creators would be reluctant to adopt it out of concern that it might degrade visual appeal, ultimately preventing disclosure), while remaining discernible to a human observer upon careful inspection on a larger display.

Overview. Figure 1 provides a concise view of our pipeline. During image generation, a diffusion model yields the PMP $\hat{x}_{0|t}$ at multiple timesteps; our core idea is to interact with these intermediate estimates sequentially using a differentiable renderer (Hong et al., 2026). Concretely, the process follows the temporal order of diffusion sampling, and each step is handled by a dedicated subsection:

1. **Locating**: update renderer parameters to determine where and how to place the stamp, using either an image-conditioned or concept-conditioned spatial prior.
2. **Blending**: adjust the image so the stamp harmonizes with the content.
3. **Filtering**: select stamps based on visibility and aesthetic quality.
4. **Embossing**: apply the final aesthetic embossing of the stamps onto the image.

Schedule. For clarity and consistency with standard diffusion notation, we describe the process over timesteps $t \in \{T, \dots, 0\}$ and define the timestep sets $\mathcal{T}_{\text{loc}} = \{t \mid t > T_\phi\}$, $\mathcal{T}_{\text{blend}} = \{t \mid T_b < t \leq T_\phi\}$, $\mathcal{T}_{\text{filter}} = \{0\}$, and $\mathcal{T}_{\text{emboss}} = \{0\}$, indicating when each operation is applied.

When concept-conditioned attribution is desired, the same schedule is applied to each detected concept instance, with locating driven by a token-specific attention prior.

4.1 LOCATING: EARLY-STAGE RENDERER PARAMETER OPTIMIZATION

4.1.1 FOR AIGC DISCLOSURE: IMAGE-CONDITIONED LOCATING

For disclosure, we optimize renderer parameters only for $t \in \mathcal{T}_{\text{loc}}$ while the image content is still in a semantic shaping stage (Esser et al., 2024; Ma et al., 2024; Park et al., 2025; Jeong et al., 2025). At the beginning of $\mathcal{T}_{\text{loc}}$, the stamp parameters $\boldsymbol{\phi}$ are initialized in a uniform grid pattern excluding the central region of the image. Each stamp is assigned a 0-degree rotation and an identical scale proportional to the image size. At each t, we compute PMP $\hat{\mathbf{x}}_{0|t}$ and render $\mathbf{r}_t = \text{Render}(\boldsymbol{\phi})$. We then update $\boldsymbol{\phi}$ by minimizing a grayscale cosine-similarity loss:

$$L_{\text{cossim}}(\mathbf{r}_t, \hat{\mathbf{x}}_{0|t}) = 1 - \cos\Big(\text{gray}(\mathbf{r}_t),\, \text{gray}(\hat{\mathbf{x}}_{0|t})\Big) \tag{3}$$

In practice, to ensure valid and aesthetic placement, we add a spatial constraint loss L_{const} to the objective, formulating the total loss as $L_{\text{total}} = L_{\text{cossim}} + \lambda L_{\text{const}}$. L_{const} comprises a boundary penalty to guide stamps within the canvas, and a repulsion penalty to prevent stamps from overlapping. Note that only the renderer parameters $\boldsymbol{\phi}$ are updated using Adam (Kingma & Ba, 2015), and $\boldsymbol{z}_t$ follows the original diffusion sampling process in this stage.

4.1.2 FOR CONCEPT-GUIDED ATTRIBUTION: CONCEPT-CONDITIONED LOCATING

For concept/IP attribution, we keep the same early-stage differentiable-renderer loop as disclosure, but change the locating prior from image-conditioned aesthetics to concept-conditioned attention. Diffusion cross-attention has been shown to provide word-to-image grounding in (Tang et al., 2023), and has been used for local editing or attention control in (Hertz et al., 2023; Chefer et al., 2023). Xie et al. (2023) further shows that cross-attention can act as a spatial constraint for generation. Inspired by these observations, we use the generator's native text-image attention as a placement cue, but we do not steer denoising or modify model parameters. The attention maps only constrain where visible stamps are placed.

For each registered target phrase (*e.g.*, 'NIKE' or 'LEGO'), we aggregate its cross-attention maps over selected layers, heads, and locating timesteps, then upsample and normalize the result into a target-specific support map. We use this map for encouraging each stamp to lie on the corresponding objects. We provide the detailed construction and losses in Appendix B.1.

Asynchronous and batched differentiable renderer execution. The concept-guided variant adds attention processing and gate-aware candidate scoring, so we use scheduling and vectorization to reduce overhead without changing the objective. We use the differentiable renderer (Hong et al., 2026) with batched FP16 calls, and lazily synchronize asynchronous renderer and image-decoder calls. For final filtering, we avoid rendering every candidate mask separately by using bounding-box luminance proxies and batch-rendering only the selected masks.

4.2 BLENDING: CONTENT HARMONIZATION

After the locating stage (§4.1), we freeze $\boldsymbol{\phi}$ (and denote the frozen render by $\boldsymbol{r}$) and apply blending for $t \in \mathcal{T}_{\text{blend}}$. We apply a lightweight blending operator to softly harmonize the stamp with the content:

$$\hat{\mathbf{x}}_{0|t} \leftarrow \hat{\mathbf{x}}_{0|t} + \text{blend}(\boldsymbol{r}). \tag{4}$$

See the supplemental materials for the detail of blend.

Latent correction for pixel-space blending. However, applying Equation (4) directly to LDMs is nontrivial. In latent diffusion, denoising occurs in the latent domain $\hat{\mathbf{z}}_{0|t}$, while $\hat{\mathbf{x}}_{0|t}$ is obtained via decoding as $\hat{\mathbf{x}}_{0|t} = \mathcal{D}(\hat{\mathbf{z}}_{0|t})$. Therefore, naïvely applying the pixel-space blending update to $\hat{\mathbf{z}}_{0|t}$ is not well-defined. It is well known that, for many LDMs, the encoder $\mathcal{E}$ is not a left-inverse of the decoder $\mathcal{D}$ (Hong et al., 2024; Chang et al., 2025). We thus adopt a correction by reformulating the approach of (Chang et al., 2025) as follows:

$$\hat{\mathbf{z}}_{0|t} \leftarrow \hat{\mathbf{z}}_{0|t} + \mathcal{E}(\mathcal{D}(\hat{\mathbf{z}}_{0|t}) + \text{blend}(\boldsymbol{r})) - \mathcal{E}(\mathcal{D}(\hat{\mathbf{z}}_{0|t})) \tag{5}$$

4.3 FILTERING: VISIBILITY–AESTHETICS SELECTION

This module selects N_{final} stamps from N_{cand} candidates based on their background grayscale luminance. Using the rendered mask $\mathbf{m}^{(i)} \in \{0,1\}^{H\times W}$ for the i-th candidate ($i = 1, \ldots, N_{\text{cand}}$), we evaluate the luminance score s_i as:

$$s_i = \frac{\sum \mathbf{m}^{(i)} \odot \text{gray}(\mathbf{x}_0)}{\sum \mathbf{m}^{(i)}}. \tag{6}$$

By filtering the candidates with the lowest scores, we prevent the intrusive pop-out effect and ensure harmonious integration.

4.4 EMBOSSING: FINAL LUMINANCE RELIEF

In practice, the most noticeable improvement in human readability comes from a final-step *luminance embossing* operation. We inject a shallow relief signal into the luminance channel only, keeping chroma nearly unchanged (Singh et al., 2025). To maintain uniform visibility, we dynamically scale the embossing strength γ for each stamp inversely to its local luminance score. The full formulation and color-space details are provided in the supplemental materials.

Why embossing matters. The embossed relief is applied to luminance only, so the mark becomes visible even when blending is weak, while chroma statistics remain largely unchanged. Empirically, this last-step operation dominates perceptual detectability without introducing large artifacts. This choice is also aligned with the *Functional Coherence* principle in visible watermark design (Ran et al., 2025): while aesthetic blending is desirable, the watermark should remain clearly recognizable as an intentional overlay rather than being mistaken as native image content. Accordingly, final embossing preserves a consistent "non-content" signal while retaining visual harmony.

4.5 EXTENSION INTO VIDEO MODEL

Extending DiffStamp to video generation requires addressing temporal consistency. While ϕ could be optimized at every frame, visual perception is acutely sensitive to motion. Continuous positional updates cause the stamp to jitter, severely degrading visual harmony and drawing unintended attention away from the generated content.

To preserve visual harmony, we freeze ϕ temporally. Filtering is executed periodically, and the selected stamps are anchored. Furthermore, as the background changes dynamically, stamps may enter or exit the scene. Abruptly adding or removing these stamps introduces jarring flicker. To ensure smooth transitions, we apply fade-in and fade-out to their embossing strengths.

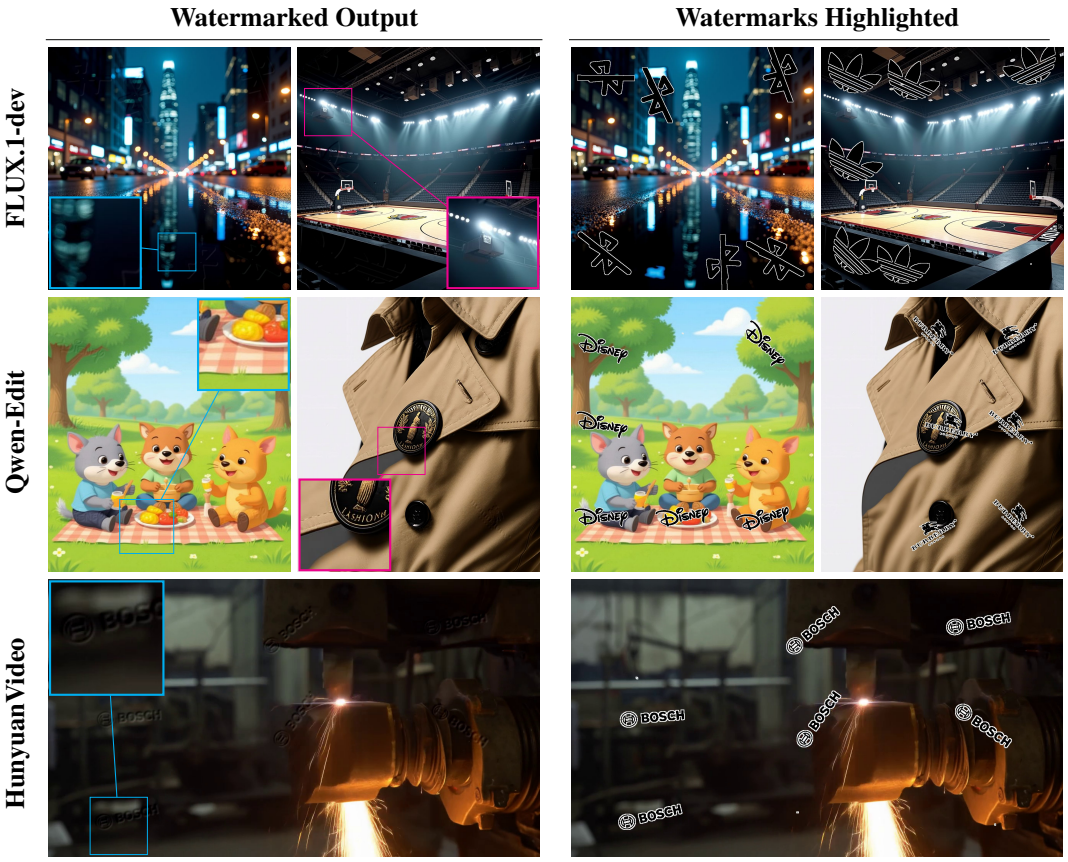
5 EXPERIMENTS

5.1 EXPERIMENTAL SETTINGS

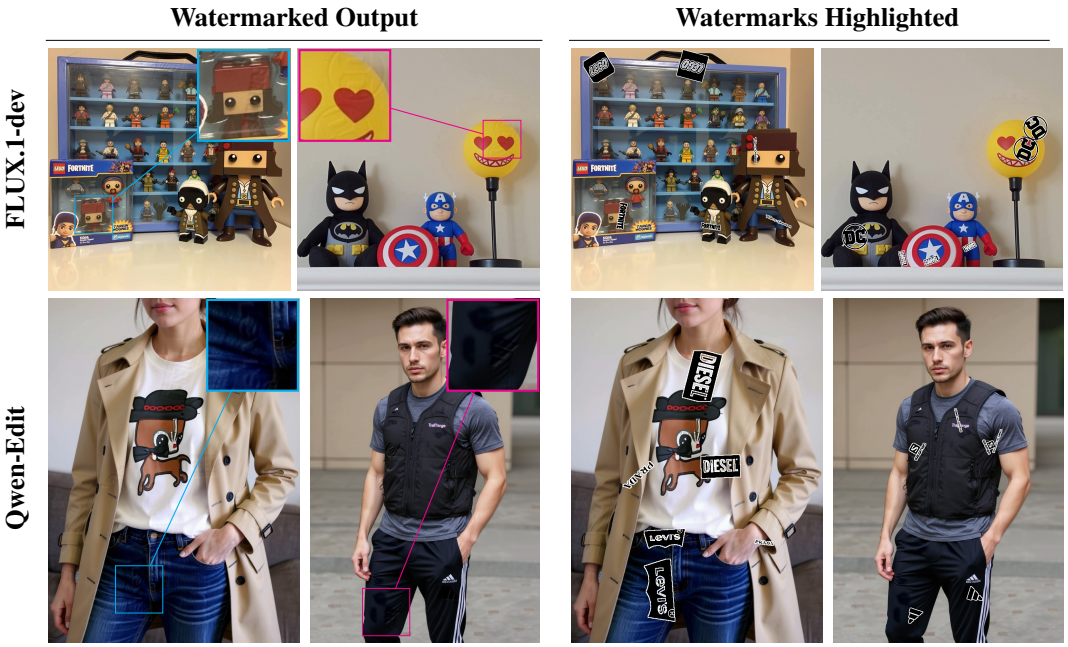
We evaluate disclosure and concept/IP attribution on three diffusion models: FLUX.1-dev (Black Forest Labs, 2024) (text-to-image), Qwen-Image-Edit (Qwen-Edit) (Wu et al., 2025a) (image-to-image), and HunyuanVideo (Kong et al., 2024) (text-to-video). We use $T = 50$ diffusion steps, $|\mathcal{T}_{\text{loc}}| = 5$ locating steps, and $|\mathcal{T}_{\text{blend}}| = 1$ blending step, with hyperparameters in Section C.1 (Table A6). For attribution, we use multi-concept prompts or edits, aggregate cross-attention over selected tokens, use $N_{\text{cand}} = 8$ initial candidates, and retain two stamps per concept under cross-concept anti-overlap. The at-scale benchmark has 165 FLUX/Qwen scenes, while HunyuanVideo is evaluated qualitatively and for runtime.

5.2 QUALITATIVE RESULTS

Figure 2a highlights the disclosure-oriented qualitative results. The stamps appear as natural stylistic elements within the scene, and the content adapts around them while remaining visually discoverable through the paired highlights. Figure 2b shows concept-guided content/IP attribution with multiple target phrase–stamp pairs per sample. Using attention-gated locating, DiffStamp places each visible



(a) Disclosure-oriented qualitative results.



(b) Concept-guided content/IP attribution qualitative results on FLUX.1-dev and Qwen-Edit.

Figure 2: **Qualitative Results. (a)** shows disclosure-oriented outputs and their watermarks-highlighted version across image and video models. **(b)** shows concept-guided content/IP attribution extension, where concept-specific stamps are assigned to different concepts within the same sample and the paired maps visualize the corresponding localized attribution regions. Additional disclosure examples are deferred to the appendix. Best viewed when zoomed in.

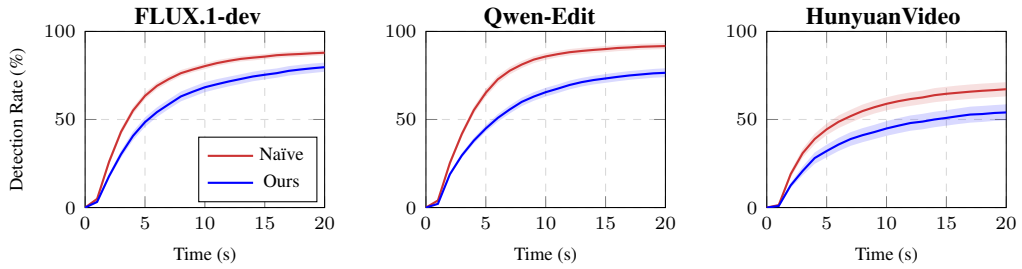


Figure 3: **Visual-search study results.** Cumulative stamp-detection rate over search time. "Naïve" denotes random overlay without locating and blending. Shaded areas show 95% confidence intervals. DiffStamp delays initial detection while remaining largely discoverable by the 20-second limit.

Table 2: **Perceptual quality and comparison with a prior visible-watermark method.** Lower is better for the 200-sample no-reference metrics. Preference rates come from a separate pairwise preference study ($N = 84$); Naïve Overlay was not included in that study.

Method	NIQE↓	ILNIQE↓	BRISQUE↓	Aesthetic↑	Less noticeable↑
Prior visible WM (Ran et al., 2025)	4.81	33.8	30.2	46.1%	27.9%
Naïve Overlay	5.04	32.5	31.0	–	–
DiffStamp (Ours)	**4.63**	**30.5**	**26.2**	**53.9%**	**72.1%**

stamp on its corresponding concept region, from toy figures and props in FLUX.1-dev to edited apparel regions in Qwen-Edit. The paired watermark visualizations show that the same formulation can be extended from a single global disclosure pattern to localized, target-conditioned attribution. For a more direct qualitative baseline comparison, Appendix C.6 compares DiffStamp against a passive post-hoc visible overlay baseline (Ran et al., 2025) on additional FLUX.1-dev and Qwen-Edit examples. Additional qualitative results are also demonstrated in the Appendix.

5.3 Human Evaluation and Perceptual Quality

Visual-search study. To measure the visibility–aesthetics trade-off, we conducted visual-search user study with 108 participants. Each participant searched for six stamps in six instances per model and method, for 36 trials in total, with a 20-second limit per trial. Figure 3 reports cumulative stamp-detection rates. Stamps placed by DiffStamp are consistently less likely to pop out during the first five seconds, while its detection rate rises throughout deliberate search. This pattern is consistent with suppressing preattentive visual pop-out (Treisman & Gelade, 1980) rather than making the disclosure inaccessible. With six stamps per sample, even the lowest 20-second rate corresponds to more than three detected stamps on average. Full protocol and interface details are in Section C.7.

Perceptual-quality and prior-method comparison. We compare against both Naïve Overlay and the closest prior visible-watermark method (Ran et al., 2025), which we re-implemented because its code was unavailable. Table 2 reports three no-reference perceptual metrics over 200 samples and a separate pairwise preference study with 84 participants. We use no-reference metrics because full-reference measures such as PSNR or LPIPS would count the intentionally visible stamp itself as an error. DiffStamp obtains the best score on all three metrics. In the pairwise preference study, each participant judges four image pairs; DiffStamp is preferred for aesthetics and, more strongly, for watermark subtlety, despite using six stamps rather than the baseline's single stamp.

5.4 Concept-aware Attribution at Scale

We evaluate attribution on 165 multi-concept FLUX and Qwen-Edit scenes, covering 396 concept instances and 792 final stamps. Text-prompted SAM 3 (Carion et al., 2025) provides usable masks for 317 instances (80%), which define the spatial-evaluation subset in Figure 4. DiffStamp consistently exceeds the corresponding random-placement and chance references. Because the stamps are much

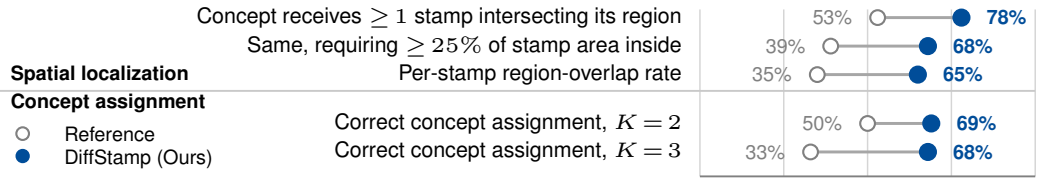


Figure 4: **Concept-aware attribution at scale.** Spatial-localization references use uniformly random same-size stamps, while concept-assignment references are chance levels of $1/K$.

smaller than their target objects, we report overlap rather than stamp-to-object IoU; the full protocol and mask-availability audit are in Section C.5.

5.5 Robustness

We evaluate DiffStamp's robustness using the regeneration attack of (Zhao et al., 2024) at three `noise_step` strengths over 200 images per method and strength. Table 3 shows that DiffStamp retains the highest post-attack detection rate throughout the sweep. Targeted visible-watermark removers (Liu et al., 2021; Niu et al., 2023; Sun et al., 2023; Mo et al., 2025; PixelBin.io, 2026) and hand-guided inpainting instead require per-image localization; representative outcomes and evaluator details are in Figures A9 and A10 and Section C.8.

Table 3: Detection rates after the regeneration attack.

`noise_step`	Naïve Overlay	No Blending	DiffStamp (Ours)
0 (no attack)	0.960	0.950	0.950
60	0.920	0.910	**0.950**
120	0.810	0.815	**0.860**
180	0.745	0.765	**0.795**

5.6 Runtime Comparison

Table 4 shows that DiffStamp introduces modest (i.e., not larger than 15%) end-to-end overhead over base models across all three backbones and in both the disclosure and attribution settings. For attribution, we report $K = 3$ with six final stamps. The full K-scaling and per-stage results are in Tables A4 and A5, and the execution-strategy ablation is in Table A2.

Table 4: **Runtime Comparison.** Each cell shows *No Stamp*→DiffStamp runtime in seconds, followed by the overhead %. The attribution row uses $K = 3$ concepts × two final stamps = total six-stamps.

Setting	FLUX.1-dev	Qwen-Edit	HunyuanVideo
Disclosure (§4.1.1)	23.32 → 25.75 **(+10.4%)**	116.4 → 120.9 **(+3.9%)**	404.6 → 462.0 **(+14.2%)**
Attribution (§4.1.2)	23.60 → 25.93 **(+9.9%)**	118.37 → 123.64 **(+4.5%)**	409.5 → 467.5 **(+14.2%)**

6 Conclusion

In this work, we present DiffStamp, a proactive visible watermarking framework for manifest AIGC disclosure. By integrating a differentiable renderer into diffusion sampling, it jointly optimizes stamp geometry and image content through locating, blending, filtering, and embossing. Objective quality metrics, user studies, and dataset-level attack sweeps across image and video backbones show model-agnostic applicability, modest overhead, and persistent yet less obtrusive visible marks. We also quantify a concept-conditioned extension for creator-side IP attribution, showing localized placement well above random baselines on 165 multi-concept scenes.

Limitations. DiffStamp is not immune to targeted editing: manual localization and inpainting can erase stamps, although they require per-image intervention. The visual-search study is controlled rather than an in-the-wild test of viewer comprehension; concept-aware attribution depends on cross-attention fidelity; and the 2D bitmap renderer does not model perspective, occlusion, or lighting, which can produce unnatural stamps on oblique, reflective, or uniformly textured surfaces. Future work will study real-world disclosure behavior and stronger semantic and 3D-aware placement priors.

GENERATIVE AI USE DISCLOSURE

In this work, we used GPT-5.2 through the OpenAI API as an automated multimodal evaluator for the regeneration-attack experiment described in Section C.8. The model was queried using a fixed prompt and structured output schema, and the reported detection rates were computed from its outputs using a predefined criterion. We also used generative AI assistants during manuscript preparation to restructure sections, draft and edit prose, format tables and references, and check LaTeX consistency. Generative AI was not used to formulate or prove mathematical claims; formal proof assistance is not applicable to this work. The authors reviewed all AI-assisted text and artifacts, checked the reported numerical values against the experimental records, and inspected the final compiled manuscript. The authors take full responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

REFERENCES

Vishal Asnani, John Collomosse, Tu Bui, Xiaoming Liu, and Shruti Agarwal. ProMark: Proactive diffusion watermarking for causal attribution. *CVPR*, 2024.

Vishal Asnani, John Collomosse, Xiaoming Liu, and Shruti Agarwal. Custommark: Customization of diffusion models for proactive attribution. *ICCV*, 2025.

Kar Balan, Shruti Agarwal, Simon Jenni, Andy Parsons, Andrew Gilbert, and John Collomosse. EKILA: Synthetic media provenance and attribution for generative art. *CVPR*, 2023.

Black Forest Labs. FLUX, 2024.

Ryan Burgert, Xiang Li, Abe Leite, Kanchana Ranasinghe, and Michael Ryoo. Diffusion illusions: Hiding images in plain sight. *SIGGRAPH*, 2024.

Nicolas Carion, Laura Gustafson, Yuan-Ting Hu, Shoubhik Debnath, Ronghang Hu, Didac Suris, Chaitanya Ryali, et al. SAM 3: Segment anything with concepts. *arXiv preprint arXiv:2511.16719*, 2025.

Pascal Chang, Sergio Sancho, Jingwei Tang, Markus Gross, and Vinicius Azevedo. LookingGlass: Generative anamorphoses via laplacian pyramid warping. *CVPR*, 2025.

Hila Chefer, Yuval Alaluf, Yael Vinker, Lior Wolf, and Daniel Cohen-Or. Attend-and-Excite: Attention-based semantic guidance for text-to-image diffusion models. *ACM Transactions on Graphics (TOG)*, 42(4), 2023.

Wenzheng Chen, Huan Ling, Jun Gao, Edward Smith, Jaakko Lehtinen, Alec Jacobson, and Sanja Fidler. Learning to predict 3d objects with an interpolation-based differentiable renderer. *NeurIPS*, 2019.

Zhen Chen, Xiuli Chai, Zhihua Gan, Binjie Wang, and Yushu Zhang. RAE-VWP: A reversible adversarial example-based privacy and copyright protection method of medical images for internet of medical things. *IEEE Internet of Things Journal*, 11(11):20757–20768, 2024.

Hung-Kuo Chu, Wei-Hsin Hsu, Niloy J Mitra, Daniel Cohen-Or, Tien-Tsin Wong, and Tong-Yee Lee. Camouflage images. *ACM Trans. Graph.*, 29(4):51–1, 2010.

Xuehao Cui, Guangyang Wu, Zhenghao Gan, Guangtao Zhai, and Xiaohong Liu. Face2QR: A unified framework for aesthetic, face-preserving, and scannable QR code generation. *NeurIPS*, 2024.

Martin de La Gorce, David J Fleet, and Nikos Paragios. Model-based 3d hand pose estimation from monocular video. *IEEE transactions on pattern analysis and machine intelligence*, 33(9): 1793–1805, 2011.

Tali Dekel, Michael Rubinstein, Ce Liu, and William T Freeman. On the effectiveness of visible watermarks. *CVPR*, 2017.

Deloitte Center for Financial Services. The rise of authorized push payment fraud, October 2024. Accessed: 2026-03-04.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. *ICML*, 2024.

European Union. Regulation (EU) 2024/1689 of the european parliament and of the council of 13 june 2024 laying down harmonised rules on artificial intelligence (artificial intelligence act). Official Journal of the European Union, L(2024/1689).

Weitao Feng, Jiyan He, Jie Zhang, Tianwei Zhang, Wenbo Zhou, Weiming Zhang, and Nenghai Yu. Catch you everything everywhere: Guarding textual inversion via concept watermarking. *arXiv preprint arXiv:2309.05940*, 2023.

Xiang Gao, Shuai Yang, and Jiaying Liu. PTDiffusion: Free lunch for generating optical illusion hidden pictures with phase-transferred diffusion model. *CVPR*, 2025.

Daniel Geng, Inbum Park, and Andrew Owens. Factorized diffusion: Perceptual illusions by noise decomposition. *ECCV*, 2024a.

Daniel Geng, Inbum Park, and Andrew Owens. Visual Anagrams: Generating multi-view optical illusions with diffusion models. *CVPR*, 2024b.

Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-or. Prompt-to-prompt image editing with cross-attention control. *ICLR*, 2023.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *NeurIPS*, 2020.

Seongmin Hong, Suh Yoon Jeon, Kyeonghyun Lee, Ernest Ryu, and Se Young Chun. Gradient-free decoder inversion in latent diffusion models. *NeurIPS*, 2024.

Seongmin Hong, Junghun James Kim, Daehyeop Kim, Insoo Chung, and Se Young Chun. DiffBMP: Differentiable rendering with bitmap primitives. *CVPR*, 2026.

Yongjian Hu and Byeungwoo Jeon. Reversible visible watermarking and lossless recovery of original images. *IEEE Transactions on Circuits and Systems for Video Technology*, 16(11):1423–1429, 2006.

Wongi Jeong, Kyungryeol Lee, Hoigi Seo, and Se Young Chun. Training-free mixed-resolution latent upsampling for spatially accelerated diffusion transformers. *arXiv preprint arXiv:2507.08422*, 2025.

Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *NeurIPS*, 2022.

Hiroharu Kato, Yoshitaka Ushiku, and Tatsuya Harada. Neural 3d mesh renderer. *CVPR*, 2018.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *ICLR*, 2015.

Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, Bo Wu, Jianwei Zhang, et al. HunyuanVideo: A systematic framework for large video generative models. *arXiv preprint arXiv:2412.03603*, 2024.

koustubhk. Popular brand Logos - Image Dataset. Accessed: 2026-03-11.

Santosh Kumar. Stylish Product Image Dataset. Accessed: 2026-03-11.

Tzu-Mao Li, Miika Aittala, Frédo Durand, and Jaakko Lehtinen. Differentiable monte carlo ray tracing through edge sampling. *ACM Transactions on Graphics (TOG)*, 37(6), 2018.

Jia-Wei Liao, Winston Wang, Tzu-Sian Wang, Li-Xuan Peng, Ju-Hsian Weng, Cheng-Fu Chou, and Jun-Cheng Chen. DiffQRCoder: Diffusion-based aesthetic QR code generation with scanning robustness guided iterative refinement. *WACV*, 2025.

Shichen Liu, Tianye Li, Weikai Chen, and Hao Li. Soft rasterizer: A differentiable renderer for image-based 3d reasoning. *ICCV*, 2019.

Tianci Liu, Tong Yang, Quan Zhang, and Qi Lei. Beyond invisibility: Learning robust visible watermarks for stronger copyright protection. In *Conference on Uncertainty in Artificial Intelligence*, pp. 2769–2785. PMLR, 2025.

Tsung-Yuan Liu and Wen-Hsiang Tsai. Generic lossless visible watermarking—a new approach. *IEEE Transactions on Image Processing*, 19(5):1224–1235, 2010.

Yang Liu, Zhen Zhu, and Xiang Bai. WDNet: Watermark-decomposition network for visible watermark removal. *WACV*, 2021.

Matthew M Loper and Michael J Black. OpenDR: An approximate differentiable renderer. *ECCV*, 2014.

Calvin Luo. Understanding diffusion models: A unified perspective. *arXiv preprint arXiv:2208.11970*, 2022.

Lykon. Dreamshaper-8 inpainting, 2023. Accessed: 2026-02-19.

Xinyin Ma, Gongfan Fang, and Xinchao Wang. Deepcache: Accelerating diffusion models for free. *CVPR*, 2024.

Anish Mittal, Anush Krishna Moorthy, and Alan Conrad Bovik. No-reference image quality assessment in the spatial domain. *IEEE Transactions on Image Processing*, 21(12):4695–4708, 2012. doi: 10.1109/TIP.2012.2214050.

Anish Mittal, Rajiv Soundararajan, and Alan C. Bovik. Making a "completely blind" image quality analyzer. *IEEE Signal Processing Letters*, 20(3):209–212, 2013. doi: 10.1109/LSP.2012.2227726.

Zihao Mo, Junye Chen, Chaowei Fang, and Guanbin Li. PatchWiper: Leveraging dynamic patch-wise parameters for real-world visible watermark removal. *ACM MM*, 2025.

Li Niu, Xing Zhao, Bo Zhang, and Liqing Zhang. Fine-grained visible watermark removal. *ICCV*, 2023.

OpenAI. ChatGPT-5.2, 2025. Accessed: 2026-03-03.

Dogyun Park, Taehoon Lee, Minseok Joo, and Hyunwoo J Kim. Blockwise flow matching: Improving flow matching models for efficient high-quality generation. *NeurIPS*, 2025.

Soo-Chang Pei and Yi-Chong Zeng. A novel image recovery algorithm for visible watermarked images. *IEEE Transactions on Information Forensics and Security*, 1(4):543–550, 2006.

PixelBin.io. WatermarkRemover.io, 2026. Accessed: 2026-03-04.

Ed Pizzi, Sreya Dutta Roy, Sugosh Nagavara Ravindra, Priya Goyal, and Matthijs Douze. A self-supervised descriptor for image copy detection. *CVPR*, 2022.

Wenfa Qi, Yuxin Liu, Sirui Guo, Xiang Wang, and Zongming Guo. An adaptive visible watermark embedding method based on region selection. *Security and Communication Networks*, 2021(1): 6693343, 2021.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. *ICML*, 2021.

Changjuan Ran, Fang Liu, Runqi Fang, Xiangyu Meng, Shenglan Cui, and Yunfan Ye. Where watermark meets beauty: Expert-guided aesthetic visible watermarking for digital artworks. *ACM MM*, 2025.

Republic of Korea. FRAMEWORK ACT ON THE DEVELOPMENT OF ARTIFICIAL INTELLIGENCE AND THE CREATION OF a FOUNDATION FOR TRUST, January 2025. Article 31, Paragraphs 2–3.

Anna Rogers. The attribution problem with generative ai, Nov 2022. URL `https://hackingsemantics.xyz/2022/attribution/`.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. *CVPR*, 2022.

Dan Ruta, Saeid Motiian, Baldo Faieta, Zhe Lin, Hailin Jin, Alex Filipkowski, Andrew Gilbert, and John Collomosse. ALADIN: All layer adaptive instance normalization for fine-grained style similarity. *ICCV*, 2021.

Vinu Sankar Sadasivan, Mehrdad Saberi, and Soheil Feizi. IConMark: Robust interpretable concept-based watermark for AI images. *arXiv preprint arXiv:2507.13407*, 2025.

Udit Sethia. Brand Logo recognition dataset. Accessed: 2026-03-11.

Gursimran Singh, Tianxi Hu, Mohammad Akbari, Qiang Tang, and Yong Zhang. Towards secure and usable 3d assets: A novel framework for automatic visible watermarking. *WACV*, 2025.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *ICLR*, 2021.

Ruizhou Sun, Yukun Su, and Qingyao Wu. Denet: Disentangled embedding network for visible watermark removal. *AAAI*, 2023.

Raphael Tang, Linqing Liu, Akshat Pandey, Zhiying Jiang, Gefei Yang, Karun Kumar, Pontus Stenetorp, Jimmy Lin, and Ferhan Türe. What the daam: Interpreting stable diffusion using cross attention. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 5644–5659, 2023.

Rohit Thanki, Surekha Borra, Vedvyas Dwivedi, and Komal Borisagar. A RONI based visible watermarking approach for medical image authentication. *Journal of medical systems*, 41(9):143, 2017.

Anne M. Treisman and Garry Gelade. A feature-integration theory of attention. *Cognitive Psychology*, 12(1):97–136, 1980. doi: 10.1016/0010-0285(80)90005-5.

Guanjie Wang, Zehua Ma, Chang Liu, Xi Yang, Han Fang, Weiming Zhang, and Nenghai Yu. MUST: Robust image watermarking for multi-source tracing. *AAAI*, 2024.

Sheng-Yu Wang, Alexei A. Efros, Jun-Yan Zhu, and Richard Zhang. Evaluating data attribution for text-to-image models. *ICCV*, 2023.

Chenfei Wu, Jiahao Li, Jingren Zhou, Junyang Lin, Kaiyuan Gao, Kun Yan, Sheng-ming Yin, Shuai Bai, Xiao Xu, Yilei Chen, et al. Qwen-image technical report. *arXiv preprint arXiv:2508.02324*, 2025a.

Guangyang Wu, Xiaohong Liu, Jun Jia, Xuehao Cui, and Guangtao Zhai. Text2QR: Harmonizing aesthetic customization and scanning robustness for text-guided QR code generation. *CVPR*, 2024.

Guangyang Wu, Huayu Zheng, Siqi Luo, Guangtao Zhai, and Xiaohong Liu. AnimateQR: Bridging aesthetics and functionality in dynamic QR code generation. *NeurIPS*, 2025b.

Jinheng Xie, Yuexiang Li, Yawen Huang, Haozhe Liu, Wentian Zhang, Yefeng Zheng, and Mike Zheng Shou. Boxdiff: Text-to-image synthesis with training-free box-constrained diffusion. *ICCV*, 2023.

Li Zhang, Shruti Agarwal, John Collomosse, Pengtao Xie, and Vishal Asnani. TokenTrace: Multi-concept attribution through watermarked token recovery. *CVPR*, 2026.

Lin Zhang, Lei Zhang, and Alan C. Bovik. A feature-enriched completely blind image quality evaluator. *IEEE Transactions on Image Processing*, 24(8):2579–2591, 2015. doi: 10.1109/TIP.2015.2426416.

Xuandong Zhao, Kexun Zhang, Zihao Su, Saastha Vasan, Ilya Grishchenko, Christopher Kruegel, Giovanni Vigna, Yu-Xiang Wang, and Lei Li. Invisible image watermarks are provably removable using generative AI. *NeurIPS*, 2024.

A Additional Experiments

A.1 Supplementary mechanism summary

Table A1 summarizes the shared mechanisms of DiffStamp for AI disclosure and creator attribution.

Table A1: Supplementary mechanism-oriented summary of DiffStamp. The same proactive watermarking components can be used for AIGC disclosure and can also be adapted to concept-conditioned creator attribution. We place this expanded side-by-side summary in the appendix because it is intended as context for the framing rather than as a standalone experimental claim.

Component view	For disclosure	For concept-guided attribution
Trigger signal	Any AIGC sample that requires manifest provenance disclosure.	Registered target phrase-stamp pairs from the prompt or edit instruction.
Spatial prior for locating	Image-conditioned cosine similarity and spatial constraints place a readable but unobtrusive stamp.	Cross-attention maps from the designated concept token provide a concept-specific spatial prior, refined by the same aesthetic constraints.
Stamp identity	A unified provenance emblem or platform-level disclosure mark.	A concept-specific emblem associated with the detected concept or asset family.
Placement granularity	Content-adaptive placement for global human readability.	Localized placement near the visual region where the concept is expressed.
Shared downstream modules	Blending, filtering, and embossing stabilize the final visible mark.	The same blending, filtering, and embossing stages are reused after concept-conditioned locating.
Intended outcome	Human-visible disclosure that the image is AI-generated.	Human-visible attribution that ties a displayed concept to its source asset or right holder.

A.2 Ablation and controllability studies

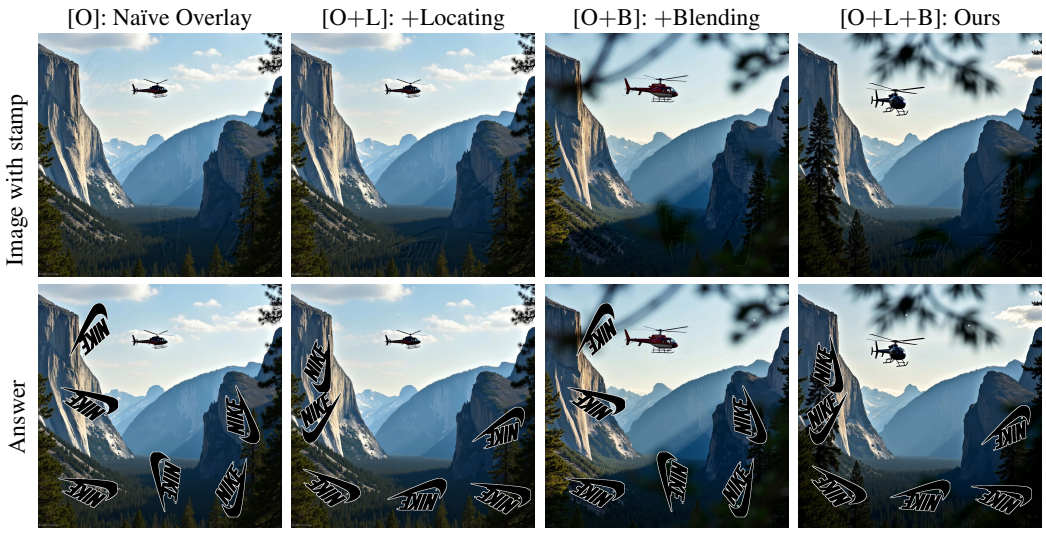


Figure A1: **Ablation study on our modules.** The watermarks are highlighted in the second row for visualization. [O], [O+B]: Without locating, the stamp may land on a highly visible region. [O], [O+L]: Without blending, the mark becomes easier to suppress under erasure attacks because the content is not harmonized with the stamp (the brightness of a specific region in [O+B], [O+L+B] is darker than in [O], [O+L]). Prompt: *"A helicopter flies over Yosemite."*

Effect of locating and blending. As demonstrated in Figure A1, both locating and blending are essential for achieving an aesthetic and persistent visible mark.

- The locating step optimizes ϕ to maximize cosine similarity with the content (Equation (3)). Without this spatial optimization, the stamp may land on low-similarity areas such as flat backgrounds, making it overly obtrusive.
- The blending step harmonizes the stamp with the content and improves persistence under removal or regeneration, as this operation is applied during the early denoising steps.

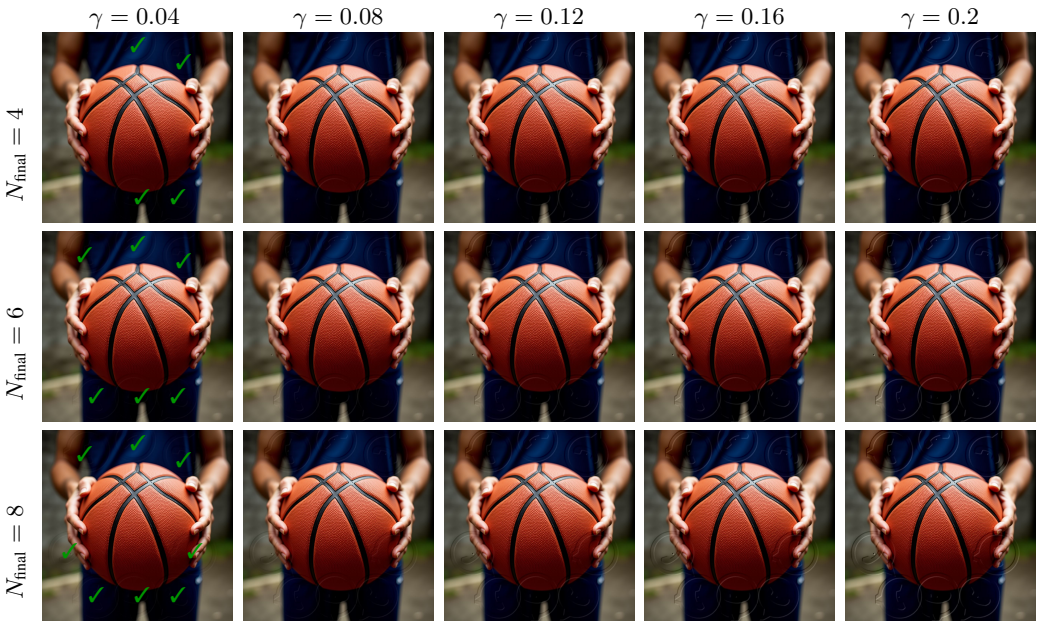


Figure A2: **DiffStamp provides flexible control over stamp visibility.** Users can tune the noticeability level they intend. N_{final} is the number of stamps per image, and γ denotes embossing strength. Answers are marked ✓(row-wise stamp locations are consistent). Toward the bottom-right, stamps become more intrusive to content, while their visibility and persistence increase. Prompt: *"the hands of a single person holding a basketball"*

Controllability. As shown in Figure A2, adjusting the embossing strength γ and the number of stamps N_{final} affects stamp visibility. Since the required visibility of a stamp may vary by application, such controllability provides practical flexibility for users. Figure A3 further demonstrates additional control options in DiffStamp. By restricting the rotation parameter, only upright stamps can be displayed.

Async & batched execution. All runtimes in this paragraph are measured in the disclosure-oriented setting; the concept/IP attribution variant is not included in this ablation. We evaluate the runtime impact of our implementation choices in differentiable renderer execution: (i) synchronous unbatched rendering, (ii) batched FP16 rendering, (iii) asynchronous launch with synchronization only when masks are consumed, and (iv) **async + batched** (ours). Table A2 shows that these optimizations substantially reduce the stamping overhead without changing the generated images, since they only affect scheduling and vectorization.

Scaling with the number of candidates. We vary the number of candidates N_{cand} and report the latency. While larger N_{cand} can improve placement refinement, the overhead grows gracefully due to batching and filtering. We report the runtime scaling in Table A3. All tests are performed in the disclosure-oriented setting on a single A100 80GB GPU.

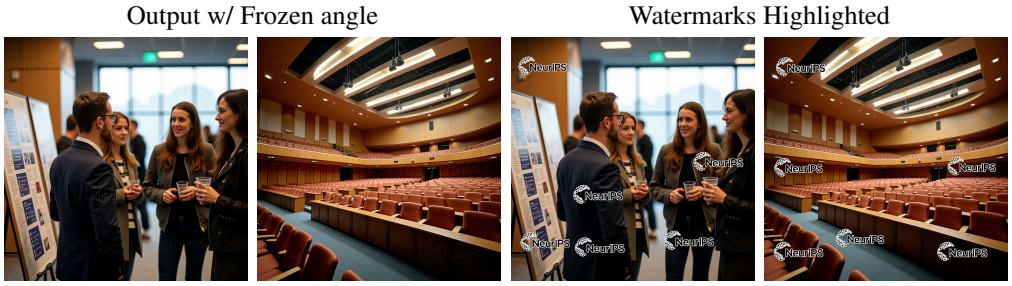


Figure A3: **Flexible control of stamp behavior.** Fixing the stamp angle improves readability. Prompt: *"Academic poster session, people talking"*, *"A large lecture hall."*

Table A2: **Runtime ablation of differentiable renderer execution strategies.** We compare (i) synchronous unbatched rendering, (ii) batched FP16 rendering, and (iii) asynchronous + unbatched execution, and (iv) asynchronous + batched execution (ours). All settings follow Section 5.1 in the disclosure-oriented setting, with $T = 50$, $|\mathcal{T}_{\text{loc}}| = 5$, $|\mathcal{T}_{\text{blend}}| = 1$, $N_{\text{cand}} = 8$, and $N_{\text{final}} = 6$. Overhead is relative to *No Stamp*.

Model	Execution	Time (s)↓	Overhead (%)↓
FLUX.1-dev	No Stamp	23.32	–
	Sync + unbatched	26.52	13.7
	Batched FP16	25.94	11.3
	Async + unbatched	26.27	12.7
	Async + batched (Ours)	25.75	10.4
Qwen-Edit	No Stamp	116.4	–
	Sync + unbatched	122.1	4.9
	Batched FP16	121.3	4.2
	Async + unbatched	121.7	4.6
	Async + batched (Ours)	120.9	3.9
HunyuanVideo	No Stamp	404.6	–
	Sync + unbatched	475.3	17.5
	Batched FP16	462.8	14.4
	Async + unbatched	474.9	17.4
	Async + batched (Ours)	462.0	14.2

A.3 Attribution Runtime Scaling

We benchmark concept-aware attribution as the number of registered concepts K increases on a single NVIDIA A100 80GB GPU. The scene and prompt are fixed per model so that differences isolate the attribution machinery. Image experiments use two warm-up runs and five measured repetitions; FLUX pools three scenes for 15 measured runs per condition. Video uses one warm-up and two measured repetitions. Each concept has eight candidates and two final stamps.

On the same HunyuanVideo scene, disclosure takes 457.2 ± 1.2s (+11.6% over no stamp), so attribution adds only 2.8/2.3/3.0% over disclosure at $K = 1/3/5$. The variation across K is comparable to the ± 1.7s run-to-run spread because attention capture is performed once per step and shared across concepts.

The profile consists of a K-independent attention-capture cost plus a per-concept increment of approximately 0.27s on FLUX and 0.28s on Qwen-Edit. On HunyuanVideo, the increment is below run-to-run variation.

Table A3: Runtime scaling with N_{cand} and rendering only $N_{\text{final}} = 6$ masks in the disclosure-oriented setting. Each entry reports end-to-end latency in seconds.

	N_{cand}			
Model	6	12	24	48
FLUX.1-dev	25.53	25.60	25.72	25.78
Qwen-Edit	120.0	120.1	120.1	120.3
HunyuanVideo	458.4	461.1	461.8	463.3

Table A4: **Attribution runtime scaling with the number of registered concepts K.** Parentheses give overhead relative to the fixed-scene no-stamp baseline.

Model	No Stamp	$K = 1$	$K = 3$	$K = 5$
FLUX.1-dev	23.60s	25.38s (+7.5%)	25.93s (+9.9%)	26.44s (+12.0%)
Qwen-Edit	118.37s	123.15s (+4.0%)	123.64s (+4.5%)	124.25s (+5.0%)
HunyuanVideo	409.5s	470.2s (+14.8%)	467.5s (+14.2%)	471.2s (+15.1%)

B Method Details

In this section, we provide detailed descriptions for each stage of our four-stage pipeline: locating, blending, filtering, and embossing.

B.1 Locating Details

First, we adapt grayscale cosine-similarity loss:

$$L_{\text{cossim}}(\mathbf{r}_t, \hat{\mathbf{x}}_{0|t}) = 1 - \cos\Big(\text{gray}(\mathbf{r}_t),\, \text{gray}(\hat{\mathbf{x}}_{0|t})\Big). \tag{A1}$$

Then we construct a constraint loss:

$$L_{\text{const}} = L_{\text{bdry}} + \lambda_r L_{\text{rep}} + \lambda_c L_{\text{cen}}. \tag{A2}$$

L_{const} combines a boundary penalty L_{bdry} (how much a stamp extends outside the $H \times W$ canvas), a repulsion penalty L_{rep} (penalizes pairwise overlap), and a center penalty L_{cen} (increases when stamps are placed farther from canvas edges). Using stamp parameters $\boldsymbol{\phi}_i = \{(x_i, y_i, r_i, \theta_i)\}$, we write

$$\begin{aligned} L_{\text{bdry}} &= \sum_i \Big[\rho(r_i - x_i) + \rho(x_i + r_i - W) \\ &\qquad + \rho(r_i - y_i) + \rho(y_i + r_i - H)\Big], \\ L_{\text{rep}} &= \sum_{i<j} \rho\big((r_i + r_j) - \|(x_i, y_i) - (x_j, y_j)\|_2\big)\,, \\ L_{\text{cen}} &= \sqrt{\frac{1}{N}\sum_i \min\{x_i,\, W - x_i,\, y_i,\, H - y_i\}^2}, \end{aligned} \tag{A3}$$

where $\rho(z) = \max(0, z)$. The total loss is

$$L_{\text{loc}} = L_{\text{cossim}} + \lambda_{\text{const}} L_{\text{const}}. \tag{A4}$$

This update is applied only within the locating window $t \in \mathcal{T}_{\text{loc}}$.

Concept-guided attention-gated locating. For concept-guided attribution, each target is specified by a target phrase and a concept-specific stamp. We aggregate the target phrase's text-image attention over selected layers, heads, and locating timesteps, upsample it to image resolution, and normalize the result to obtain a single phrase-level support map. For multi-token phrases, the constituent token maps are first aggregated into one phrase map before this normalization so that each target contributes one support map regardless of token length.

Table A5: CUDA kernel time per generation at $K = 5$ (ms). The kernels overlap asynchronously with denoising, so columns need not sum to the wall-clock difference in Table A4.

Stage	FLUX.1-dev	Qwen-Edit	HunyuanVideo
Attention capture (shared)	373	1160	12218
Map and gate updates	339	267	72
Locating renderer	307	376	453
Final gated selection	40	42	52

For multi-target prompts, we optionally reweight overlapping target maps by target-wise soft assignment:

$$P_k(x) = \mathrm{softmax}_k(A_k(x)/\tau), \qquad M_k(x) = \mathrm{Norm}\left(A_k(x)((1-\beta) + \beta P_k(x))\right). \tag{A5}$$

This reweighting uses only the model's own attention maps and does not require external boxes or masks.

We use M_k as a soft semantic support map. The gate G_k is obtained by smoothing M_k, thresholding positive values at a quantile, and applying lightweight morphology such as dilation, closing, and erosion. The same gate is shared by all candidates assigned to target k in a given sample. Candidate centers are seeded from high-attention peaks of M_k, and G_k constrains the subsequent renderer update.

For any spatial map S and nonnegative weight map w, define

$$\langle S \rangle_w = \frac{\sum_x w(x) S(x)}{\sum_x w(x) + \epsilon}. \tag{A6}$$

For the i-th candidate of target k, the target-support loss is

$$\mathcal{L}_{\mathrm{sup}}^{(k)} = -\langle M_k \rangle_{m_i} + \lambda_{\mathrm{ov}}[\eta - \langle G_k \rangle_{m_i}]_+^2 + \lambda_{\mathrm{ctr}}[1 - G_k(p_i)]_+^2. \tag{A7}$$

The first term attracts the stamp to the target support. The second term keeps the stamp mask inside the gate. The third term keeps the stamp center inside the gate. When p_i is not on an integer pixel, $G_k(p_i)$ is obtained by bilinear sampling.

The visual regularizer is evaluated inside the gate:

$$\mathcal{R}_{\mathrm{vis}}^{(k)} = \langle Y \rangle_{m_i G_k} - \lambda_{\mathrm{tex}} \langle T \rangle_{m_i G_k}, \tag{A8}$$

where Y is background luminance and T is a normalized local texture map. Thus, semantic placement is determined by M_k and G_k, while luminance and texture act as local aesthetic tie-breakers.

When heatmap seeding is enabled, the anchor term keeps each candidate near its initial attention-seeded position:

$$\mathcal{R}_{\mathrm{anc}}^{(k)} = \frac{\|p_i - \bar{p}_i\|_2^2}{\bar{r}_i^2 + \epsilon}, \tag{A9}$$

where $\bar{p}_i$ and $\bar{r}_i$ are the seeded center and radius. Boundary and repulsion constraints are inherited from the disclosure locating objective.

At final selection, candidates are scored with respect to their target gate. Candidates outside G_k receive a large penalty. Valid candidates are ranked by in-gate luminance and texture, filtered by NMS, and passed to the same blending, filtering, and embossing stages as in the disclosure setting.

Confidence-gated fallback. Because a concept attention map can be diffuse or empty, we treat spatial association as conditional rather than guaranteed. We measure the normalized entropy of the positive support and check whether the refined gate is nonempty. If the map fails either confidence check, the concept gate is bypassed and placement reverts to the global disclosure rule: luminance-based scoring over the full canvas followed by NMS. The concept emblem remains visible, but the resulting stamp is not counted as a gate-validated localization. This prevents a degenerate attention map from silently producing a false spatial attribution claim. Calibration results are reported in Section C.5.

B.2 Blending Details

We apply a blending operator to softly harmonize the stamp with the content:

$$\hat{\mathbf{x}}_{0|t} \leftarrow \hat{\mathbf{x}}_{0|t} + \mathrm{blend}(\mathbf{r}), \tag{A10}$$

we use Gaussian blur and a mean correction:

$$\mathrm{blend}(\mathbf{r}) = \mathrm{clip}_{[-1,1]}(-\alpha\,\mathbf{m} + \beta\,\mu)\,, \tag{A11}$$

where $\mathbf{m} \in [0,1]^{H\times W}$ is a soft mask after Gaussian blur from $\mathbf{r}$, $\mu = \mathrm{mean}(\alpha\,\mathbf{m})$, and α, β denote blending rate and mean reduction factor. $\beta > 0$ prevents the mean of $\hat{\mathbf{x}}_{0|t}$ from dropping too much.

The soft mask $\mathbf{m}$ is generated using a 7x7 Gaussian kernel with $\sigma = 3.0$. The blending residual is computed in pixel space, strictly clipped to the [-1, 1] range to match the VAE boundaries, and then inserted into the latent correction term in Equation (5). This update is applied only within the blending window $t \in \mathcal{T}_{\text{blend}}$.

B.3 Filtering Details

At filtering time, we render the candidate masks $\mathbf{m}^{(i)}$ and compute the local luminance score $s_i = \frac{\sum \mathbf{m}^{(i)} \odot \mathrm{gray}(\mathbf{x}_0)}{\sum \mathbf{m}^{(i)}}$. Candidates with lower scores are preferred because they produce less intrusive visible marks. To prevent the final selected stamps from visually clustering or overlapping, we then apply Non-Maximum Suppression (NMS) on the candidate bounding boxes with an Intersection over Union (IoU) threshold of 0.3, and keep the lowest-score surviving candidates for the final embossing stage.

B.4 Embossing Details

This section provides the full formulation of the luminance embossing step. Let $\mathbf{x} \in [-1,1]^{3\times H\times W}$ be the final solver state and $\mathbf{l} \in [0,1]^{H\times W}$ be the grayscale logo on the canvas.

(1) Smoothed logo and emboss map. We smooth the logo with a Gaussian filter G_σ and compute Sobel derivatives:

$$\tilde{\mathbf{l}} = G_\sigma * \mathbf{l}, \qquad \partial_x\tilde{\mathbf{l}} = S_x * \tilde{\mathbf{l}}, \qquad \partial_y\tilde{\mathbf{l}} = S_y * \tilde{\mathbf{l}}. \tag{A12}$$

Assuming a top-left light direction, the emboss map is

$$\mathbf{e} = -\big(\partial_x\tilde{\mathbf{l}} + \partial_y\tilde{\mathbf{l}}\big). \tag{A13}$$

(2) Convert to luminance-chroma space. We map $\mathbf{x}$ to $\mathbf{x}^{(0,1)} = (\mathbf{x}+1)/2$ and compute (Y, C_b, C_r) as

$$\begin{aligned} Y &= 0.299R + 0.587G + 0.114B, \\ C_b &= -0.1687R - 0.3313G + 0.5B, \\ C_r &= 0.5R - 0.4187G - 0.0813B. \end{aligned} \tag{A14}$$

(3) Emboss luminance and reconstruct RGB. We update only luminance with embossing strength γ:

$$Y' = \mathrm{clip}_{[0,1]}\big(Y + \gamma\,\mathbf{e}\big). \tag{A15}$$

Reconstruction uses the approximate inverse transform:

$$R' = Y' + 1.402\,C_r, \quad G' = Y' - 0.344136\,C_b - 0.714136\,C_r, \quad B' = Y' + 1.772\,C_b, \tag{A16}$$

and we map back to $[-1,1]$ via $\mathbf{x}' = 2\,[R', G', B']^\top - 1$. Additionally, to maintain uniform visual perceptibility across varying backgrounds, the embossing strength is dynamically normalized per stamp using the local luminance score s_i.

$$\gamma = \max(0.5, 1 - s_i)\,\gamma_{\text{max}} \tag{A17}$$

Table A6: Model-specific hyperparameters used in DiffStamp.

Hyperparameter	FLUX.1-dev	Qwen-Edit	HunyuanVideo
Inference steps T	50	50	50
$\lvert\mathcal{T}_{\text{loc}}\rvert$	5	5	5
Inner updates U	2	2	2
Learning rate (x, y)	20.0	20.0	20.0
Learning rate (r)	0.4	0.4	0.4
Learning rate (θ)	0.4	0.4	0.4
$\lvert\mathcal{T}_{\text{blend}}\rvert$	1	1	1
α	0.2	0.3	0.01
β	0.2	0.2	0.0
N_{cand}	8	8	8
N_{final}	6	6	6
Filtering period (frame)	–	–	40
γ_{max}	0.08	0.08	0.04
Fade-in/out length (frame)	–	–	20

Table A7: Input prompts for representative examples used across the experiments. In concept-guided attribution examples, colored text marks the target phrases used for concept-specific stamp localization, with different colors indicating different targets.

Figure	**Model**	**Prompt**
Figure 2	FLUX.1-dev	*"A modern glass skyscraper reflected in a puddle on a rainy night, neon city lights, sharp focus"* *"A basketball court in an empty arena, dramatic lighting, realistic"* *"Collectible display featuring: LEGO minifigure case; Fortnite Peely figure; Jack Sparrow figure."* *"Figure-and-plush display on a plain shelf: Captain America shield toy; Batman signal lamp."*
	Qwen-Edit	*"Cute animated-style animals having a picnic in a bright park, cheerful colors, children's book illustration vibe, clean composition"* *"the same logo as a photorealistic luxury fashion emblem engraved on a trench coat button, professional fashion product photography, bright softbox lighting, ultra detailed materials"* *"Transform the input image into a realistic scene: Editorial fashion photo featuring: Atelier sleeveless trench coat; Threadmark cotton t-shirt; Denim House straight jeans."* *"Transform the input image into a realistic scene: Editorial fashion photo featuring: TrailForge running vest; TriStripe track pants; DotAtelier shirt."*
	HunyuanVideo	*"A workshop interior, sparks flying and mechanical parts moving rapidly."*
Figure A9	FLUX.1-dev	*"a futuristic city"* *"a stop sign with a large tree behind it"*
Figure A10	FLUX.1-dev	*"Pilot girl in a snowstorm"*

C Detailed Experimental Settings

C.1 DiffStamp Parameters

Table A6 summarizes the parameters in DiffStamp. Filtering period and fade-in/out length are used only for the video model discussed in Section 4.5.

C.2 Base Models

FLUX.1-dev. FLUX.1-dev (Black Forest Labs, 2024) is a flow-matching-based text-to-image latent diffusion model trained at large scale and known for strong generation quality. The model has

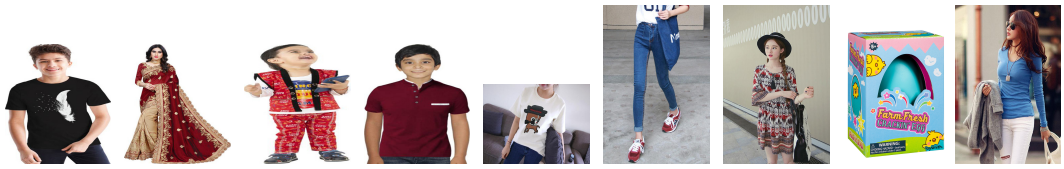


Figure A4: Reference images used for Qwen-Edit.

approximately 12B parameters. In our experiments, we generate 1024×1024 images using $T = 50$ inference steps with a guidance scale of 3.5.

Qwen-Image-Edit. Qwen-Image-Edit (Wu et al., 2025a) (Qwen-Edit) is a flow-matching-based diffusion transformer model for image-to-image editing, trained at large scale for strong instruction-following and visual fidelity. The model has approximately 20B parameters. In our experiments, we use $T = 50$ inference steps with a CFG scale of 4.0.

HunyuanVideo. HunyuanVideo (Kong et al., 2024) is a 13B-parameter latent diffusion video generation model built on a DiT backbone and trained with flow matching. In our experiments, we use $T = 50$ denoising steps with a generation setting of 544×960 resolution and 65 frames.

C.3 Qualitative Results Details

Here, we provide the corresponding input prompts for the images from the main paper and the supplementary qualitative figures in Table A7. For the Qwen-Edit image-to-image qualitative examples, the reference images and watermark assets used in our experiments are shown in Figure A4. The reference images were collected from publicly available Kaggle datasets: *Stylish Product Image Dataset* (Kumar) for product images, and *Popular brand Logos - Image Dataset* (koustubhk) and *Brand Logo recognition dataset* (Sethia) for watermark images. These images were used only as reference inputs for qualitative visualization.

C.4 Objective Quality and Pairwise-Baseline Protocol

We compute NIQE (Mittal et al., 2013), ILNIQE (Zhang et al., 2015), and BRISQUE (Mittal et al., 2012) on 200 samples produced by each of the three compared methods: the re-implemented prior visible-watermark method (Ran et al., 2025), Naïve Overlay, and DiffStamp. All three metrics are no-reference natural-image-quality estimators, and lower scores indicate better perceptual quality. We choose them over full-reference distances because there is no watermark-free image that differs only in a nuisance variable: the visible mark and the content adaptation around it are intentional parts of the output.

For the complementary pairwise preference study, 84 participants each evaluate four pairs from the prior method and DiffStamp. For every pair, participants answer two questions: which image is more aesthetic and which image contains the less noticeable watermark. The prior method places one watermark per image, whereas the default DiffStamp setting places six. Aggregate preferences are reported in Table 2.

C.5 Attribution Evaluation Details

The attribution benchmark contains 165 scenes (82 FLUX and 83 Qwen-Edit), divided into 99 two-concept and 66 three-concept scenes. This yields 396 concept instances. We retain two final stamps per concept, for 792 stamps in total. HunyuanVideo is not included in this spatial-accuracy benchmark.

We use text-prompted SAM 3 (Carion et al., 2025) to derive a concept-region mask for each instance. A usable mask is available for 317 of 396 instances (80%), and spatial metrics are computed over these 317. The remaining 79 instances are not silently discarded as placement failures: a manual audit finds that they primarily reflect the concept not being cleanly generated or being occluded, or the

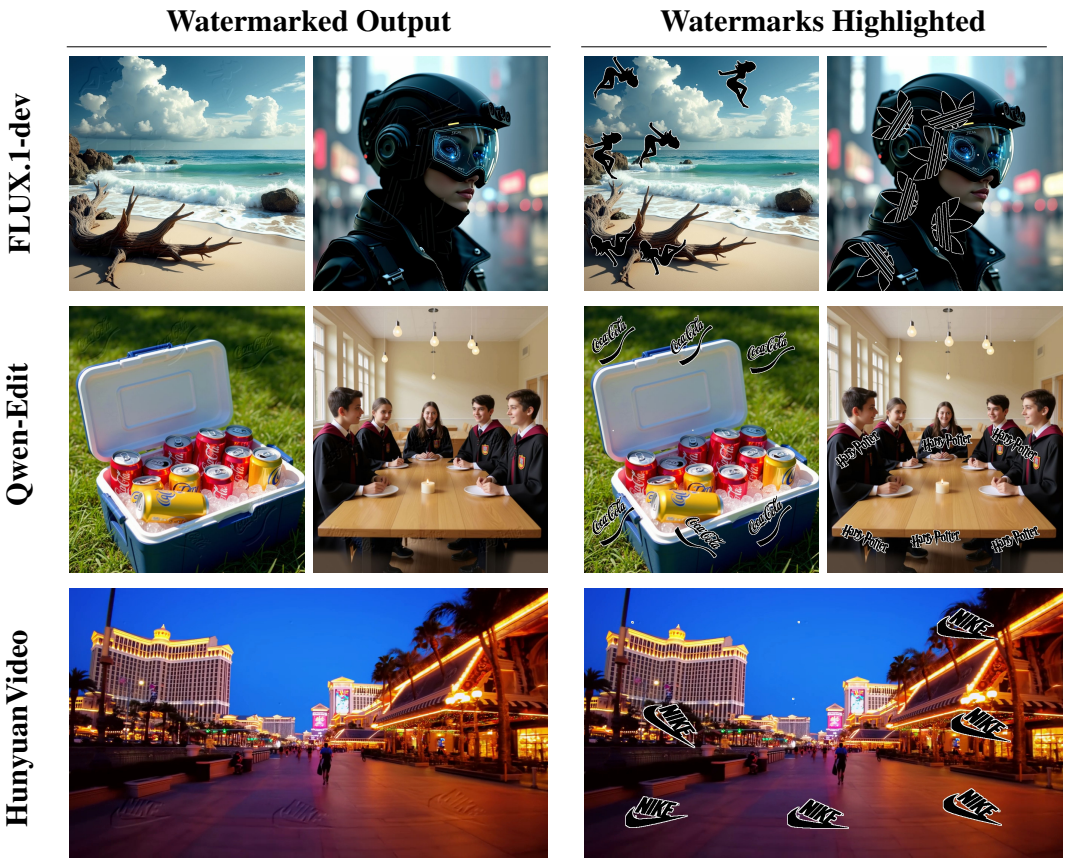


Figure A5: Additional qualitative disclosure results. The left half shows extra outputs, and the right half highlights the watermarks, including both representative HunyuanVideo frames.

segmenter missing fictional brand names despite the object being visible. Nevertheless, SAM 3 masks are proxy annotations rather than human-drawn ground truth, and this is a source of evaluation noise.

A permissive hit occurs when a stamp bounding box intersects its concept mask. The stricter hit requires at least 25% of the stamp area to lie inside the mask. We avoid stamp-to-object IoU because stamps are deliberately much smaller than the objects, which bounds IoU near zero even for a semantically correct placement. For the random-placement reference, we place same-size stamps uniformly at random 500 times per instance; the per-concept rows use the best of two random stamps, matching our method. Assignment accuracy asks whether the best-overlapping stamp is associated with the correct concept, with chance $1/K$ for K concepts.

The confidence-gated fallback described in Section B.1 is deliberately conservative. Its confidence test fires on 0 of the 396 benchmark attention maps, including 0 of the 317 independently scorable instances. We additionally test a non-visual concept phrase and a scene in which the requested concept object is absent; both maps remain above the concentration threshold, and the adaptive quantile gate still produces a feasible region. Thus, in the present benchmark the entropy test behaves as a last-resort guard for degenerate maps rather than a frequent execution path. A fallback stamp preserves visible disclosure but is not treated as a validated localized attribution.

C.6 Visual comparison with post-hoc overlay method

To further demonstrate the aesthetic benefits of our proactive framework, we provide a qualitative comparison between a passive post-hoc overlay baseline (Ran et al., 2025) and DiffStamp. As discussed in the main text, naive post-hoc visible watermarking simply pastes a mark onto an already generated image. This approach lacks any awareness of the underlying visual structure, often resulting

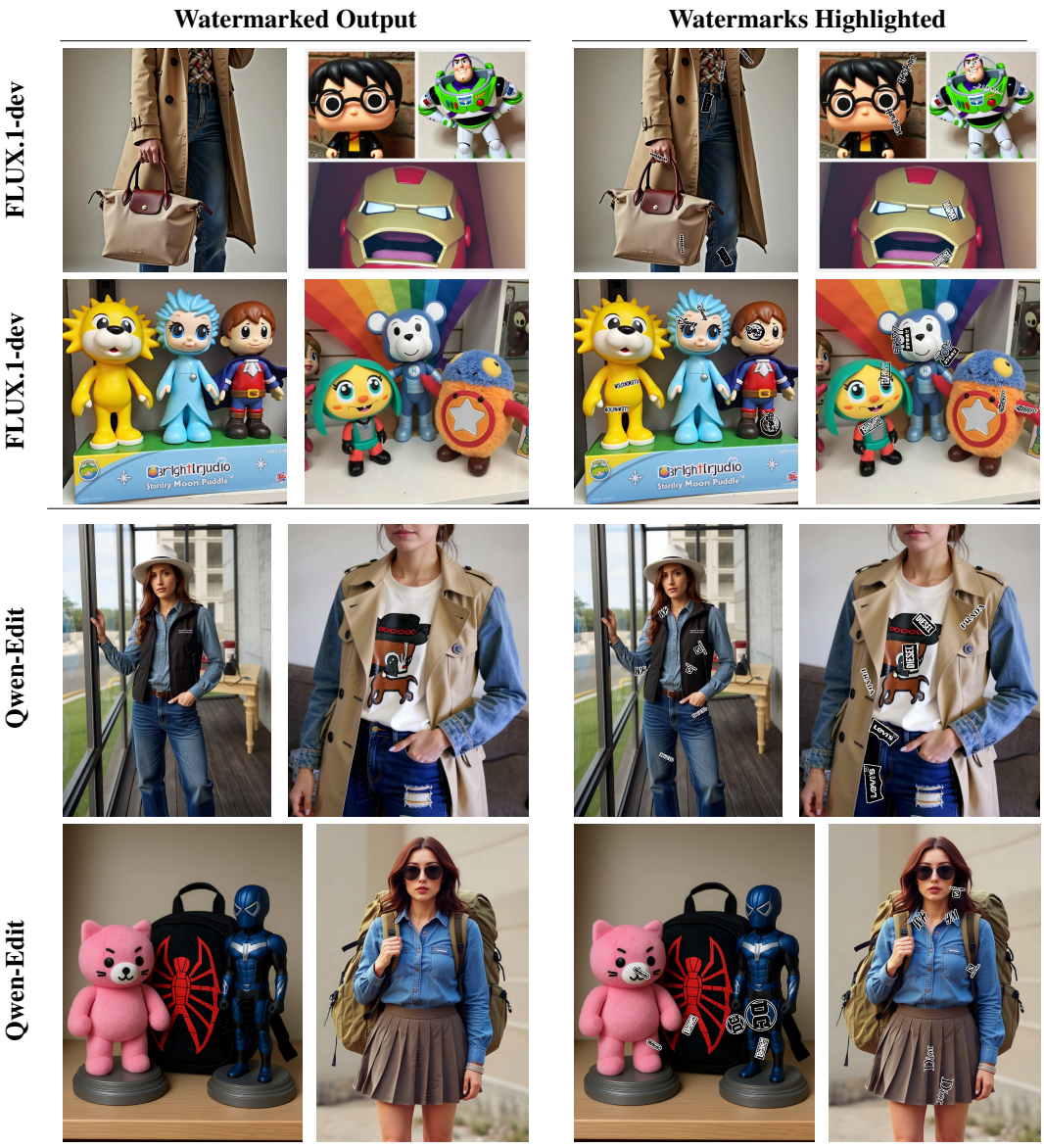
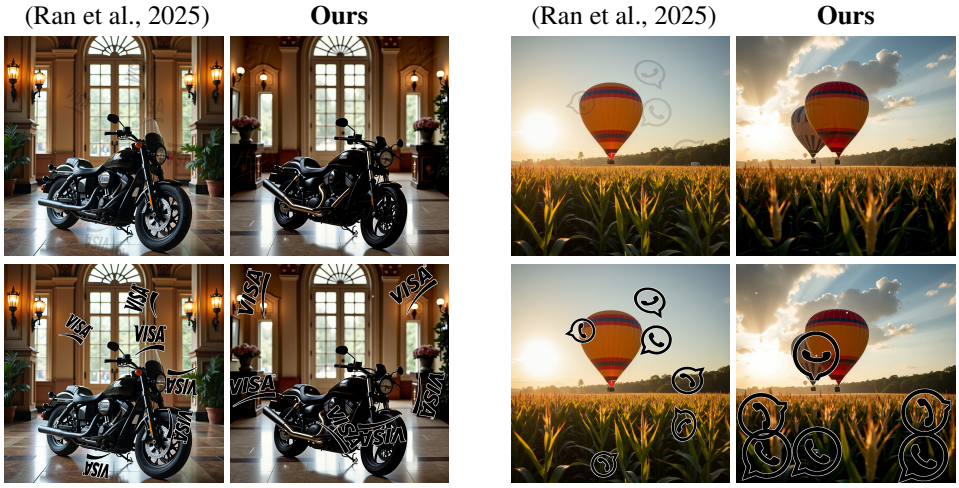


Figure A6: More qualitative results of concept-guided content/IP attribution, where concept-specific stamps are assigned to different concepts within the same sample and the paired maps visualize the corresponding localized attribution regions. Best viewed when zoomed in.

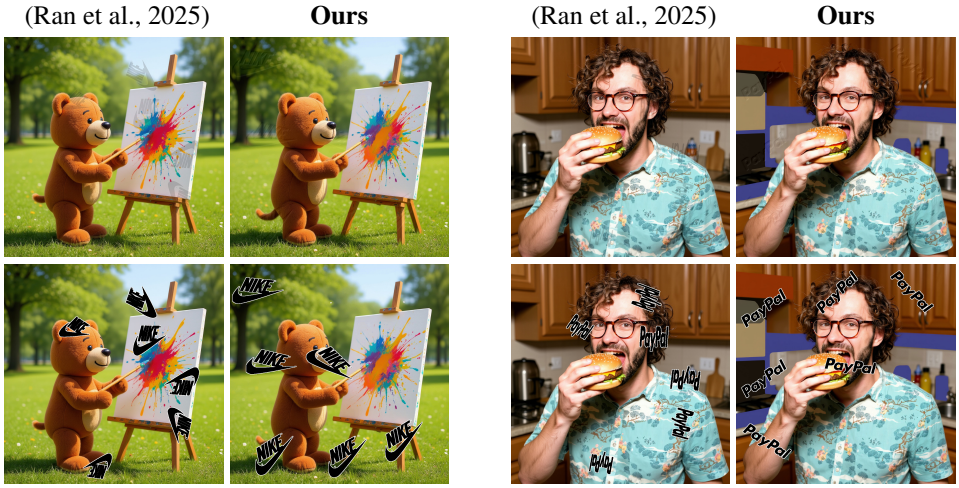
in intrusive visual artifacts that overlap with critical subjects or disrupt the image's harmony. In contrast, as shown in Figure A7, our method successfully harmonizes the stamps with the surrounding content, achieving a subtle, aesthetically pleasing integration that avoids the jarring "pop-out" effect of the post-hoc baseline.

C.7 Visual Search Study Details

Participant demographics. The visual-search user study involved 108 participants. Because every participant evaluated both Naïve Overlay and DiffStamp in a paired within-subject design, stable participant-level differences are shared across the two conditions. We do not conduct demographic subgroup analysis, so generalization across populations remains part of the broader limitation of a controlled visual-search study.



(a) Results on FLUX.1-dev.



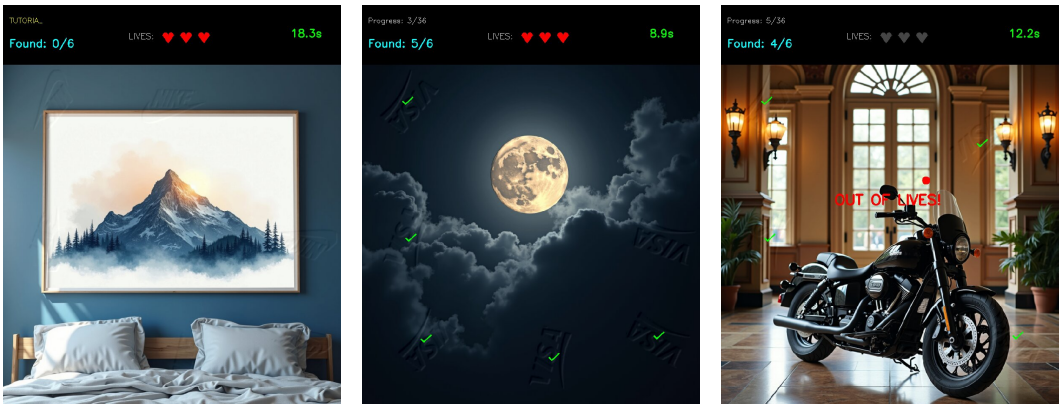
(b) Results on Qwen-Edit.

Figure A7: **Qualitative comparison of Post-hoc overlay (Ran et al., 2025) vs. DiffStamp (Ours).** Each subfigure displays two distinct examples. For each example, the left column represents the naive post-hoc overlay, while the right column shows our DiffStamp result. The top row of each subfigure displays the generated outputs, and the bottom row highlights the watermarks. Unlike the post-hoc baseline which heavily disrupts the scene, DiffStamp organically integrates the marks into the aesthetic structure of the image. Prompt: *"The bear painting a colorful splash on a canvas in a sunny park."*, *"The man is eating hamburger in the kitchen."*

Participant risk note. The visual-search study involved only a short task on generated images. The analyzed study data consisted of task responses and timestamps, while demographic information is reported only in aggregate. We did not obtain formal IRB or equivalent review for this study.

UI and instructions. The evaluation interface was designed for participants to find and click the six hidden stamps in each image as quickly as possible. Before the visual-search trials, participants completed 1 practice session to familiarize themselves with the task. Figure A8 shows the actual UI used in the experiment. Participants were instructed to locate the stamps within a 20-second time limit.

Randomization protocol. The study consisted of six instances per model and method (Naïve vs. Ours), corresponding to 36 trials per participant. For HunyuanVideo, the six instances were derived



(a) Tutorial screen (b) Evaluation screen (c) Penalty screen

Figure A8: **Visual-search study interface.** The top bar shows progress, the number of stamps found so far, and the remaining lives and time. (a) The tutorial screen familiarizes participants with the task. (b) The main evaluation screen where participants search for hidden stamps within a 20-second limit. A check mark appears when a stamp is found. (c) The "Out of lives" message is displayed when a participant makes three incorrect clicks.

by selecting two frames from each of three videos. The model order was fixed (FLUX.1-dev $\rightarrow$ Qwen-Edit $\rightarrow$ HunyuanVideo), while image and method order within each model were randomized for each participant.

Timing measurement rule. The detection time was measured for exactly 20 seconds from the moment the image was fully loaded and rendered on the screen. Each time a participant clicked on a perceived stamp, the timestamp was recorded in milliseconds. To prevent blind guessing, a penalty system was enforced: if a participant made three incorrect clicks, they lost their remaining time for that instance, and the interface automatically transitioned to the next image.

Confidence-Interval computation details. The intervals in Figure 3 are 95% confidence intervals computed at each time point from participant-level cumulative detection rates as $1.96(\sigma/\sqrt{n})$, where σ is the sample standard deviation and $n = 108$ participants.

C.8 Robustness Experiment Details

Attack method. We use PixelBin watermark remover (PixelBin.io, 2026) as a practical targeted remover and adopt the regeneration attack of (Zhao et al., 2024) at `noise_step`$\in \{60, 120, 180\}$. The lowest value is the attack's default setting; the larger values provide progressively stronger regeneration.

Removal cost and threat model. DiffStamp improves persistence under the tested automated regeneration attacks, but it is not immune to targeted editing. As shown in Figure A10, manual localization plus a strong inpainting model can erase stamps, shifting removal from scalable automation to per-image intervention.

Evaluator setup. In the robustness test summarized in Table 3, we employ a multimodal large language model as an automated evaluator. Specifically, GPT-5.2 (OpenAI, 2025) inspects images before and after attack. For every method configuration (Naïve Overlay, No Blending, and Ours) and every attack strength, we evaluate 200 generated images and their attacked counterparts. Wilson score intervals are computed independently for each post-attack detection rate.

Prompt design. To assess whether the watermark remains visible or leaves identifiable traces post-attack, we assigned the VLM a "Quality Assurance (QA) inspector" persona. The model was

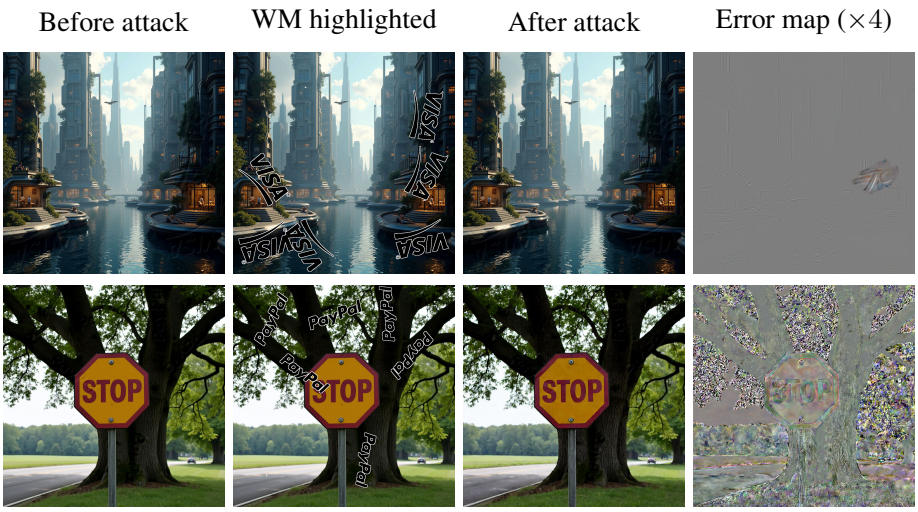


Figure A9: **Representative automated attack outcomes.** PixelBin's visible-watermark remover leaves a visible trace (top), while regeneration designed for invisible watermarks (Zhao et al., 2024) changes fine details without fully erasing the stamp (bottom). Dataset-level results are in Table 3.



(a) DiffStamp output (b) Manual masking (c) Inpainting result (d) Error map (×4)

Figure A10: **Targeted hand-guided inpainting.** Manual localization followed by Dreamshaper Inpainting V8 (Lykon, 2023) can erase a stamp, but requires per-image intervention.

informed that 6 marks were originally embedded, and it was instructed to look for both intact marks and degraded remnants. The exact prompt provided to the model was as follows:

> *"You are a meticulous Quality Assurance (QA) inspector for digital image generation.*
> *IMPORTANT CONTEXT: There were exactly 6 embedded marks (such as logos or stamps) originally placed in this image. Due to the nature of the generative process or image compression, some of these marks might be clearly visible, while others might be heavily degraded, appearing as blurry patches, smudges, or texture distortions.*
> *Your task is to carefully scan the entire image and find all 6 locations where a mark was placed.*
> *Please analyze the image based on the following:*
> *1. Clear Marks: Are there any distinct and clearly readable logos, text, or stamps?*
> *2. Degraded Marks: Are there any smudged, heavily blurred patches, or unnatural texture distortions that look like the remnants of a degraded mark?*
> *3. Count (Out of 6): Knowing there were originally 6 marks, how many distinct locations can you find in total that contain EITHER a clear mark OR a degraded mark? (Return an integer from 0 to 6).*
> *Provide your judgment purely based on careful visual inspection."*

Detection criterion and scoring protocol. The MLLM was constrained to output its response strictly following a predefined schema, which included boolean flags (`has_clear_mark` and `has_degraded_mark`) and an integer count (`detected_count` up to 6). Consistent with our definition of successful provenance disclosure in a real-world scenario, a trial was considered a

Algorithm 1 DiffStamp four-stage pipeline with latent-diffusion timestep scheduling

Input: initial latent $\mathbf{z}_T$, denoiser ϵ_θ, decoder $\mathcal{D}$, encoder $\mathcal{E}$, renderer $\mathrm{Render}(\boldsymbol{\phi})$, timestep set $\{T, \dots, 0\}$
Schedule: $\mathcal{T}_{\mathrm{loc}} = \{t \mid t > T_\phi\}$, $\mathcal{T}_{\mathrm{blend}} = \{t \mid T_b < t \le T_\phi\}$, $\mathcal{T}_{\mathrm{filter}} = \{0\}$, $\mathcal{T}_{\mathrm{emboss}} = \{0\}$
Hyperparams: inner locating updates U, candidates N_{cand}, selected stamps N_{final}
Init: initialize candidate stamp parameters $\boldsymbol{\phi} = \{\boldsymbol{\phi}_i\}_{i=1}^{N_{\text{cand}}}$ on a grid (center-excluded)
for $t = T, T-1, \dots, 1$ **do**
    // DDIM PMP prediction in latent space
    $\boldsymbol{v}_t \leftarrow \epsilon_\theta(\mathbf{z}_t, t)$
    $\hat{\mathbf{z}}_{0|t} \leftarrow (\mathbf{z}_t - \sqrt{1-\bar{\alpha}_t}\,\boldsymbol{v}_t)/\sqrt{\bar{\alpha}_t}$
    $\hat{\mathbf{x}}_{0|t} \leftarrow \mathcal{D}(\hat{\mathbf{z}}_{0|t})$
    // Stage 1: Locating ($t \in \mathcal{T}_{\mathrm{loc}}$)
    if $t \in \mathcal{T}_{\mathrm{loc}}$ **then**
        $\mathbf{r}_t \leftarrow \mathrm{Render}(\boldsymbol{\phi})$
        for $u = 1, \dots, U$ **do**
            $L_{\mathrm{loc}} \leftarrow L_{\mathrm{cossim}}(\mathbf{r}_t, \hat{\mathbf{x}}_{0|t}) + \lambda_{\mathrm{const}} L_{\mathrm{const}}(\boldsymbol{\phi})$
            $\boldsymbol{\phi} \leftarrow \mathrm{AdamStep}(\boldsymbol{\phi}, \nabla_{\boldsymbol{\phi}} L_{\mathrm{loc}})$
            if $u < U$ **then**
                $\mathbf{r}_t \leftarrow \mathrm{Render}(\boldsymbol{\phi})$
            end if
        end for
    end if
    // Stage 2: Blending ($t \in \mathcal{T}_{\mathrm{blend}}$)
    if $t \in \mathcal{T}_{\mathrm{blend}}$ **then**
        Freeze $\boldsymbol{\phi}$ and set $\mathbf{r} \leftarrow \mathrm{Render}(\boldsymbol{\phi})$
        $\hat{\mathbf{z}}_{0|t} \leftarrow \hat{\mathbf{z}}_{0|t} + \mathcal{E}\big(\mathcal{D}(\hat{\mathbf{z}}_{0|t}) + \mathrm{blend}(\mathbf{r})\big) - \mathcal{E}\big(\mathcal{D}(\hat{\mathbf{z}}_{0|t})\big)$
        $\hat{\mathbf{x}}_{0|t} \leftarrow \mathcal{D}(\hat{\mathbf{z}}_{0|t})$
    end if
    // sampler step
    $\mathbf{z}_{t-1} \leftarrow \sqrt{\bar{\alpha}_{t-1}}\,\hat{\mathbf{z}}_{0|t} + \sqrt{1-\bar{\alpha}_{t-1}}\,\boldsymbol{v}_t$
end for
$\mathbf{x}_0 \leftarrow \mathcal{D}(\mathbf{z}_0)$
// Stage 3: Filtering ($t = 0 \in \mathcal{T}_{\mathrm{filter}}$)
for $i = 1, \dots, N_{\text{cand}}$ **do**
    Render binary mask $\mathbf{m}^{(i)}$ and score $s_i \leftarrow \frac{\sum \mathbf{m}^{(i)} \odot \mathrm{gray}(\mathbf{x}_0)}{\sum \mathbf{m}^{(i)}}$
end for
Apply NMS on candidate bounding boxes with IoU threshold 0.3 and keep the lowest-score surviving indices $\mathcal{I}_{\text{final}}$
// Stage 4: Embossing ($t = 0 \in \mathcal{T}_{\mathrm{emboss}}$)
for $i \in \mathcal{I}_{\text{final}}$ **do**
    Set emboss strength $\gamma_i \leftarrow \max(0.5, 1 - s_i)\,\gamma_{\max}$
    $\mathbf{x}_0 \leftarrow \mathrm{Emboss}_Y\big(\mathbf{x}_0, \mathbf{m}^{(i)}, \gamma_i\big)$
end for
Return: final stamped image/video frame $\mathbf{x}_0$

"successful detection" if the VLM flagged at least one mark as either clear or degraded. The overall detection rate was computed by dividing the number of successfully detected images by the total number of trials (200 images). This protocol was applied identically to both the pre-attack and post-attack image sets to measure the drop in detectability.

D Algorithm

Algorithm 1 shows our four-stage pipeline for latent diffusion models.

E Broader Impacts

DiffStamp may have positive societal impact by making synthetic media more legible as such and by giving creators a lightweight way to attach attribution cues at generation time. In particular, proactive visible marks can support AI-manifest disclosure, reduce accidental misrepresentation of AI-generated content, and offer an opt-in path for creator-side attribution or consent signaling when

protected styles, logos, or concepts are involved. At the same time, visible marks can be misused as false provenance claims, may be applied without the consent of depicted people or brand owners, and can interfere with faces, text, or other accessibility-critical content if placed carelessly. These risks motivate transparent deployment policies, verifiable provenance pipelines, and placement controls that avoid sensitive regions.